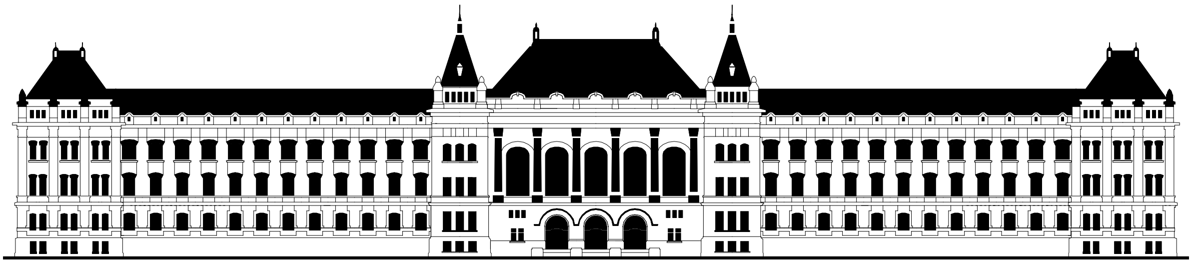
IEEE Hungary Section
Circuits, Systems and Computers Joint Chapter
Instrumentation and Measurement & Engineering
in Medicine and Biology Joint Chapter

PROCEEDINGS
OF THE
20TH PHD MINI-SYMPOSIUM

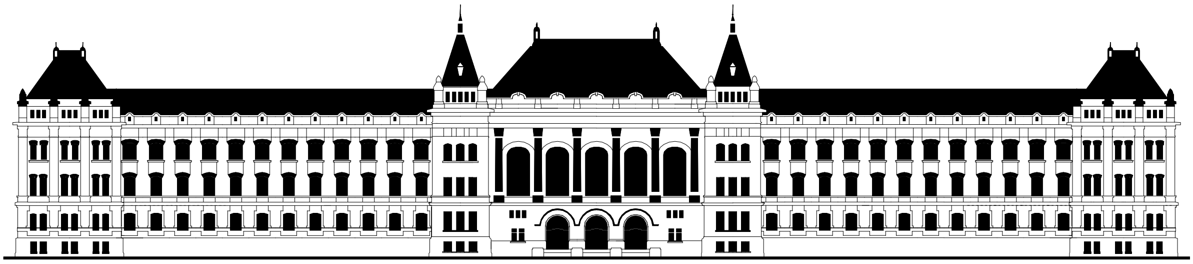
FEBRUARY 4, 2013.



Budapest University of Technology and Economics
Department of Measurement and Information Systems

PROCEEDINGS
OF THE
20TH PHD MINI-SYMPOSIUM

FEBRUARY 4, 2013.
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
BUILDING I



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
DEPARTMENT OF MEASUREMENT AND INFORMATION SYSTEMS

Head of the Department: Prof. Dr. Ákos JOBBÁGY

Conference Chairman:
Béla PATAKI

Organizers:
Dorottya CSERPÁN
Kristóf GÁTI
Péter GYÖRKE
Áron HORVÁTH

Homepage of the Conference:
http://minisy.mit.bme.hu/

Sponsored by:
IEEE Hungary Section (technical sponsorship)
Schnell László Foundation

ISBN 978-963-593-503-1

FOREWORD

This proceedings is a collection of the lectures of the 20th PhD Mini-Symposium held at the Department of Measurement and Information Systems of the Budapest University of Technology and Economics. The main purpose of these symposiums is to give an opportunity to the PhD students of our department to present a summary of their work done in the preceding year. It is an interesting additional benefit, that the students get some experience: how to organize such events. Beyond this actual goal, it turned out that the proceedings of our symposiums give an interesting overview of the research and PhD education carried out in our department. The lectures reflect partly the scientific fields and work of the students, but we think that an insight into the research and development activity of the department is also given by these contributions. Traditionally our activity was focused on measurement and instrumentation. The area has slowly changed during the last few years. New areas mainly connected to embedded information systems, new aspects e.g. dependability and security are now in our scope of interest as well. Both theoretical and practical aspects are dealt with.

The proceedings will not be published in printed form, it has turned out that nowadays the web publication of symposium lectures is enough. This new form has some advantages, but it has some challenges as well. We hope that the advantages will dominate.

The papers of this proceedings could be sorted into some main groups. These are Embedded and Intelligent Systems; Measurement and Signal Processing; Model-based Software Engineering and Knowledge Representation. The lectures are at different levels: some of them present the very first results of a research, others contain more new results. Some of the first year PhD students have been working on their fields only for half a year, therefore they submit two-page papers. The second and third year students are more experienced and have more results; therefore they have four-page papers in the proceedings. There are two types of papers in the proceedings. One is a short independent publication; the other is simply a summary of work. This second one is intended to give an overview of the work done during the last year; therefore it could contain shorter or longer parts of the PhD student's other publication; and it does not necessarily contain new results, which have not been published earlier.

During this twenty-year period there have been shorter or longer cooperation between our department and some universities, research institutes, organizations and firms. Some PhD research works gained a lot from these connections. In the last year the cooperation was especially fruitful with the Vrije Universiteit Brussel Dienst ELEC, Belgium; Innomed Medical Zrt., Budapest; IEEE Instrumentation and Measurement Society & Engineering in Medicine and Biology Society Joint Chapter, IEEE Hungary Section.

We hope that similarly to the previous years, also this PhD Mini-Symposium will be useful for the lecturers, for the audience and for all who read the proceedings.

Budapest, January , 2013.

Béla Pataki
Chairman of the PhD Mini-Symposium

LIST OF PARTICIPANTS

Participant	Advisor	Starting Year of PhD Course
BOLGÁR, Bence	ANTAL, Péter	2012
CSERPÁN, Dorottya	HORVÁTH, Gábor - SOMOGYVÁRI, Zoltán	2011
DEMIÁN, Tamás	PATARICZA, András	2010
DÜLK, Ivor	KOVÁCSHÁZY, Tamás	2012
GALAMBOS, Róbert	SUJBERT, László	2010
GYÖRKE, Péter	PATAKI, Béla	2011
HORVÁTH, Áron	HORVÁTH, Gábor	2011
IZSÓ, Benedek	RÁTH, István - VARRÓ, Dániel	2012
JÁNOSI, András	DABÓCZI, Tamás	2012
PÁLFI, Vilmos	KOLLÁR, István	2010
SZIKRA, István	FEHÉR, Béla	2012

Program of The MINI-SYMPOSIUM

Embedded Systems — Chair: JOBBÁGY, Ákos

Model-based Software Technology and Intelligent Systems — Chair: MAJZIK, István

Conference Schedule

Time	February 4, 2013
8:30	Embedded Systems
10:10	Coffee break
10:40	Model-based Software Technology and Intelligent Systems

A FAST FOUR-PARAMETER SINE WAVE ESTIMATION METHOD[1] (ANNUAL RESEARCH REPORT)

Vilmos PÁLFI
Advisor: István KOLLÁR

I. Introduction

This work is an annual research report, based on the parts of paper [5]. The author would like to suggest reading paper [5] instead of this lecture.

Accurate estimation of sine wave parameters is an important field of signal processing. The recommended method by the IEEE standard for ADC testing [1] is the four parameter fitting algorithm, using the time domain samples. However, this method could be very time consuming for high number of samples. In this paper an alternative, frequency domain method is presented with the capability to provide the sine parameters much faster, without significant loss of precision.

II. Estimation of sine parameters

A Implementation

To speed up the fitting algorithm, the frequency domain samples are used instead of the time domain samples. To compress the information in the frequency domain, the four-term Blackman-Harris window is used which has side lobes under -91 dB [2]. The time domain expression of the window is [2]:

$$w(k) = a_0 + a_1 \cdot \cos\left(\frac{2\pi k}{N}\right) + a_2 \cdot \cos\left(\frac{4\pi k}{N}\right) + a_3 \cdot \cos\left(\frac{6\pi k}{N}\right) \qquad (1)$$

where

$$a_0 = 0.35875, a_1 = -0.48829, a_2 = 0.14128, a_3 = -0.01168. \qquad (2)$$

If we want to use the formula of a windowed sine wave in the frequency domain, first we need the expression of the nonwindowed *N* length sine (which takes leakage into account):

$$X_{sin}(k) = X_{sin}^{-}(k) + X_{sin}^{+}(k) \qquad (3)$$

$$X_{sin}^{-}(k) = e^{-j\pi(k-f)\frac{N-1}{N}} \cdot \frac{A+jB}{2} \cdot \frac{\sin\left(\pi(k-f)\right)}{\sin\left(\pi(k-f)\frac{1}{N}\right)} \qquad (4)$$

$$X_{sin}^{+}(k) = e^{-j\pi(k+f)\frac{N-1}{N}} \cdot \frac{A-jB}{2} \cdot \frac{\sin\left(\pi(k+f)\right)}{\sin\left(\pi(k+f)\frac{1}{N}\right)} \qquad (5)$$

where

$$A = R \cdot \cos(\varphi)\,, B = R \cdot \sin(\varphi) \qquad (6)$$

and the expression of the sine wave in the time domain is

$$x(t) = R \cdot cos(2\pi f_x t + \varphi) \qquad (7)$$

where R, f_x, ϕ are the amplitude, frequency and initial phase of the sine wave, respectively. In the sine fitting procedure the following formulas are used to avoid large arguments in the input of the trigonometric functions (and as a consequence of large arguments, roundoff errors):

$$f = \lfloor f \rfloor + \langle f \rangle, f_i = \lfloor f \rfloor, f_f = \langle f \rangle \qquad (8)$$

$$X_{sin}^{-} = e^{j\pi\left(f_f + \frac{k-f}{N}\right)} \cdot \frac{A+jB}{2} \cdot \frac{\sin\left(-f_f\pi\right)}{\sin\left(\frac{k-f}{N}\pi\right)} \qquad (9)$$

$$X_{sin}^{+} = e^{j\pi\left(-f_f + \frac{k+f}{N}\right)} \cdot \frac{A-jB}{2} \cdot \frac{\sin\left(f_f\pi\right)}{\sin\left(\frac{k+f}{N}\pi\right)} \qquad (10)$$

---

[1] This one year research report is based on the parts of [5].

where f_i and $\lfloor f \rfloor$ is the integer part and f_f and $\langle f \rangle$ is the fractional part of f, which is the number of periods the sine wave contained.

If the measured signal contains a dc component, the mathematical model has to be modified:

$$x(t) = R \cdot \cos\left(\frac{2\pi f n}{N} + \varphi\right) + C \tag{11}$$

where C is the level of the dc component and n=0,1,…,N-1. This means that an X_{dc} part has to be added to the X_{sin} part in the frequency domain to represent the signal correctly:

$$X(k) = DFT\left\{R \cdot \cos\left(\frac{2\pi f n}{N} + \varphi\right) + C\right\} \tag{12}$$

$$X(k) = X_{dc}(k) + X_{sin}(k) \tag{13}$$

where

$$X_{dc}(k) = C \cdot N, if\ k = 0 \pm N \pm 2 \cdot N \pm 3 \cdot N \pm \cdots \tag{14}$$

and

$$X_{dc}(k) = 0\ otherwise \tag{15}$$

If the signal is windowed with samples of the four term Blackman-Harris window, the multiplication in the time domain becomes a convolution in the frequency domain:

$$X_{BH}(k) = DFT\{x(t) \cdot w(t)\} = \underline{a}^T \cdot \underline{y} \tag{16}$$

$$\underline{a} = \left[\frac{a_3}{2}\ \frac{a_2}{2}\ \frac{a_1}{2}\ a_0\ \frac{a_1}{2}\ \frac{a_2}{2}\ \frac{a_3}{2}\right]^T \tag{17}$$

$$\underline{y}^T = [X(k-3), X(k-2), X(k-1), X(k), X(k+1), X(k+2), X(k+3)]. \tag{18}$$

Above expression is linear in the amplitude, initial phase (more precisely in the terms A and B which are related to R and ϕ), and the dc component. However, it is nonlinear in the frequency. This means that during the fitting an iterative numerical method has to be used. For this purpose, the Newton-Gauss method was chosen because it has two advantageous properties [3]:

- The second order derivatives are neglected in the Hesse-matrix,
- The positive semi-definiteness of the Hesse-matrix is ensured.

The Newton-Gauss method was performed on the following cost-function:

$$K = \frac{1}{2}\underline{e}^T \underline{e} \tag{19}$$

where e is the vector of residuals. The definition of the Newton-Gauss step is

$$\Delta\underline{p} = (\boldsymbol{J}^T \boldsymbol{C}^{-1} \boldsymbol{J}) \boldsymbol{J}^T \boldsymbol{\Sigma}^{-1} \underline{e} \tag{20}$$

where $\underline{p}$, $\boldsymbol{J}$, $\boldsymbol{\Sigma}$ are the vector of parameters (A, B, C, f), the so-called Jacobian matrix and the covariance matrix, respectively. The Jacobian matrix has a size N×4 and consists of the derivatives with respect to the parameters A, B, C and f.

$$\boldsymbol{J} = \left[\frac{\partial X_{BH}}{\partial A}, \frac{\partial X_{BH}}{\partial B}, \frac{\partial X_{BH}}{\partial C}, \frac{\partial X_{BH}}{\partial f}\right] \tag{21}$$

This is calculated using the derivatives of the expression

$$X(k) = DFT\left\{R \cdot \cos\left(\frac{2\pi f t}{N} + \varphi\right) + C\right\} \tag{22}$$

$$\frac{\partial X(k)}{\partial A} = \frac{\partial X_{sin}^{-}(k)}{\partial A} + \frac{\partial X_{sin}^{+}(k)}{\partial A} \tag{23}$$

$$\frac{\partial X(k)}{\partial B} = \frac{\partial X_{sin}^{-}(k)}{\partial B} + \frac{\partial X_{sin}^{+}(k)}{\partial B} \tag{24}$$

$$\frac{\partial X(k)}{\partial f} = \frac{\partial X_{sin}^{-}(k)}{\partial f} + \frac{\partial X_{sin}^{+}(k)}{\partial f} \tag{25}$$

$$\frac{\partial X(k)}{\partial C} = \frac{\partial X_{dc}(k)}{\partial C} \tag{26}$$

$$\frac{\partial X_{sin}^{-}(k)}{\partial A} = e^{j\pi\left(f_f + \frac{k-f}{N}\right)} \cdot \frac{1}{2} \cdot \frac{\sin(-f_f \pi)}{\sin\left(\frac{k-f}{N}\pi\right)} \tag{27}$$

$$\frac{\partial X_{sin}^{+}(k)}{\partial A} = e^{j\pi\left(-f_f + \frac{k+f}{N}\right)} \cdot \frac{1}{2} \cdot \frac{\sin(f_f \pi)}{\sin\left(\frac{k+f}{N}\pi\right)} \tag{28}$$

$$\frac{\partial X_{sin}^{-}(k)}{\partial B} = e^{j\pi\left(f_f+\frac{k-f}{N}\right)} \cdot \frac{j}{2} \cdot \frac{\sin\left(-f_f\pi\right)}{\sin\left(\frac{k-f}{N}\pi\right)} \tag{29}$$

$$\frac{\partial X_{sin}^{+}(k)}{\partial B} = e^{j\pi\left(-f_f+\frac{k+f}{N}\right)} \cdot \frac{-j}{2} \cdot \frac{\sin\left(f_f\pi\right)}{\sin\left(\frac{k+f}{N}\pi\right)} \tag{30}$$

$$\begin{gathered}\frac{\partial X_{sine}^{-}(k)}{\partial f} = j\pi\frac{N-1}{N} \cdot X_{sin}^{-}(k) + e^{j\pi\left(f_f+\frac{k-f}{N}\right)} \cdot \frac{A+jB}{2} \cdot \ldots \\ \cdot \frac{-\pi\cdot\cos\left(f_f\pi\right)\cdot\sin\left(\frac{k-f}{N}\pi\right)+\frac{\pi}{N}\cos\left(\frac{k-f}{N}\pi\right).\sin\left(-f_f\pi\right)}{\frac{1-\cos\left(\frac{2\pi(k-f)}{N}\right)}{2}}\end{gathered} \tag{31}$$

$$\begin{gathered}\frac{\partial X_{sin}^{+}(k)}{\partial f} = -j\pi\frac{N-1}{N} \cdot X_{sin}^{+}(k) + e^{j\pi\left(-f_f+\frac{k+f}{N}\right)} \cdot \frac{A-jB}{2} \cdot \ldots \\ \cdot \frac{\pi\cdot\cos\left(f_f\pi\right)\cdot\sin\left(\frac{k+f}{N}\pi\right)-\frac{\pi}{N}\cos\left(\frac{k+f}{N}\pi\right).\sin\left(f_f\pi\right)}{\frac{1-\cos\left(\frac{2\pi(k+f)}{N}\right)}{2}}\end{gathered} \tag{32}$$

$$\frac{\partial X_{dc}(k)}{\partial C} = N \text{ for } k = 0 \pm N \pm 2 \cdot N \pm 3 \cdot N \pm \cdots \tag{33}$$

$$\frac{\partial X_{dc}(k)}{\partial C} = 0 \text{ otherwise} \tag{34}$$

Using the above expressions the derivatives of X_{BH} can be calculated as

$$\frac{\partial X_{BH}(k)}{\partial A} = \underline{a}^T \cdot \frac{\partial \underline{y}^T}{\partial A}, \frac{\partial X_{BH}(k)}{\partial B} = \underline{a}^T \cdot \frac{\partial \underline{y}^T}{\partial B} \tag{35}$$

$$\frac{\partial X_{BH}(k)}{\partial f} = \underline{a}^T \cdot \frac{\partial \underline{y}^T}{\partial f}, \frac{\partial X_{BH}(k)}{\partial C} = \underline{a}^T \cdot \frac{\partial \underline{y}^T}{\partial C} \tag{36}$$

Since the model (16) is nonlinear in parameter *f*, it is important to estimate accurately its value to ensure the convergence of the Newton-Gauss method. For this purpose, interpolated FFT was used where the initial guess for the frequency was calculated using the analytic solution for the rectangular windows (for more details see [4]).

Using the above described method, the fit can be done using only the points of the Blackman-Harris windows around the sine frequency and the dc frequency, because the significant part of the information about the sine is compressed into these points. During the fitting algorithm 35 points are used (12 points represent the dc part and 23 the sine). This way only the time consumption of the FFT depends on the number of samples, the computational burden of the fit does not. This is an important difference between this and the original fitting algorithm which uses the time domain samples, and results in a significant increase of speed compared to the original method.

B Properties of the estimator

Since the distribution of the noise in the time domain is unknown (a measured sine may contain Gaussian noise, harmonic distortion, quantization noise, etc.), as an approximation we can use the Central Limit Theorem, and assume that the noise in the frequency domain is more or less Gaussian since on every frequency it is the linear combination of a noise samples which has unknown distribution. Then the properties of the least-squares estimator will be approximately the following:

- the estimator is asymptotically normally distributed,
- the estimator is asymptotically unbiased (the mean value of the estimation error is 0),
- the covariance matrix of the estimator asymptotically reaches the Cramer-Rao lower bound.

III. Comparison of the new and the original algorithm

A Precision and accuracy

In this section the estimation errors of the algorithm are compared from the point of view of the mean value and variance. In these simulations a sine disturbed by Gaussian noise (zero mean, *LSB*/3 standard deviation) was quantized with a nonideal 10 bit ADC. The INL and DNL characteristics of the quantizer can be seen in Fig. 1. 10000 records were created where the amplitude, frequency, dc component and initial phase were random variables. The parameters of each record were estimated with both sine fitting procedures, and then the statistical properties were compared. Table I. shows the results where μ_t is the mean value of the original estimator (which uses time domain samples), μ_f is the mean value of the new estimator (which uses frequency domain samples), σ_t is the standard deviation of the original estimator and σ_f, is the standard deviation of the new estimator.

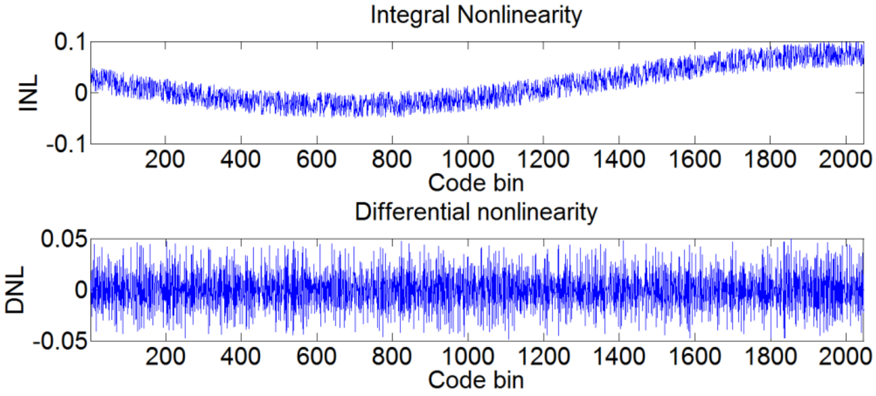


Figure 1: INL and DNL of 10 bit nonideal quantizer

$\hat{A}$		$\hat{B}$		$\hat{C}$		$\hat{f}$	
μ_t/μ_f	σ_t/σ_f	μ_t/μ_f	σ_t/σ_f	μ_t/μ_f	σ_t/σ_f	μ_t/μ_f	σ_t/σ_f
0,788	0,996	0,914	0,988	1,000	0.993	0,377	0,949

Table I: Comparison of statistical properties for each parameter

B *Comparison of speed*

Table II. shows the different computational times of the algorithms for different record lengths where t_t is the time requirement of the original method and t_f is of the new method (both in seconds). The time consumption of the FFT in the new method is included in these results.

2^{14} samples		2^{16} samples		2^{18} samples		2^{20} samples		2^{22} samples	
t_t	t_f	t_t	t_f	t_t	t_f	t_t	t_f	t_t	t_f
0.11	0.09	0.45	0.10	1.98	0.15	7.73	0.31	14.95	0.64

Table II: Comparison of computational burden

IV. Conclusion

In this lecture an alternative fitting method was presented. It was shown that it is able to provide the sine parameters without significant loss in the precision but much faster than the original algorithm.

References

[1] "IEEE Standard for Terminology and Test Methods for Analog-to-Digital Converters," IEEE Std 1241-2010 (Revision of IEEE Std 1241-2000) , vol., no., pp.1-139, Jan. 14 2011.

[2] Harris, F.J.; , "On the use of windows for harmonic analysis with the discrete Fourier transform," Proceedings of the IEEE , vol.66, no.1, pp. 51- 83, Jan. 1978.

[3] Adriaan van den Bos (2007), “Parameter Estimation for Scientists and Engineers”, Wiley-Interscience.

[4] Bilau, T. Z.; Megyeri, T.; Sárhegyi, A.; Márkus, J; Kollár, I.; , “Four-parameter fitting of sine wave testing result: iteration and convergence,” Computer Standards and Interfaces, vol. 26, pp. 51-56, 2004.

[5] Palfi, V.; Kollar, I.; , "Efficient execution of ADC test with sine fitting with verification of excitation signal parameter settings," Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International , vol., no., pp.2662-2667, 13-16 May 2012

THE ADVANTAGES OF THE WIIR FILTER IMPLEMENTATION COMPARED TO IIR FILTER IMPLEMENTATIONS

Róbert GALAMBOS
Advisors: László SUJBERT, Balázs BANK

I. Introduction

Most commonly all IIR filters are designed on a linear frequency scale, however there are several cases where the application of the filter would suggest another approach. E.g. the human ear has an almost logarithmic resolution, therefore it is better for audio applications to use nearly logarithmic scale. Several studies have been published on warped frequency scale processing. Besides warping discrete sine and cosine transform, warping IIR (WIIR) filters reveal great potential especially for audio applications [1, 2, 3].

II. Warping IIR Filters

The idea of the WIIR filter is simple. By replacing the z^{-1} delays in an IIR filter to $H(z)$ all-pass filters the WIIR scheme is obtained. The all-pass filter can be seen as a frequency dependent (dispersive) delay. The phase response of the all-pass filter will delay each frequencies differently causing the dispersive behaviour. The simplest and most commonly used all-pass filter is the first order all-pass filter. It corresponds to the following bilinear mapping:

$$z^{-1} \Rightarrow \frac{-\lambda + z^{-1}}{1 - \lambda z^{-1}} \tag{1}$$

where λ sets the phase response and so the warping of the frequency scale. Mostly this parameter is set to meet other well known auditory scales. By setting $\lambda \approx 0.74...0.78$ the Bark and Greenwood scale is approached.

A. Delay Free Loops

The first order all-pass filter as dispersive delay can not be used in its original form. The WIIR scheme will contain delay free loops that make the filter unimplementable in the give form. This problem can be solved by reorganising the filter equation to eliminate these delay free loops [4]:

$$x_0 = \frac{1}{\alpha_0}\left(In - \sum_{i=1}^{N+1} \alpha_i x_{i-1} z^{-1}\right) \tag{2}$$

where α_i coefficients are calculated from the original a_i coefficients using the following equations [5]:

$$\alpha_0 = a_0 + \sum_{i=1}^{N} (-\lambda)^i a_i \tag{3}$$

$$\alpha_1 = \sum_{i=1}^{N} (-\lambda)^{i-1} a_i \tag{4}$$

$$\alpha_k = \sum_{i=k}^{N} (-\lambda)^{i-k} a_i - \sum_{i=k-1}^{N} (-\lambda)^{i-k+2} a_i \tag{5}$$

where $k = 2..N + 1$.

B. *Implementable WIIR filters*

Derived from (1) a section of a WIIR filter can be implemented without delay free loops as seen in Figure 1 (lines to the next or from previous stage are marked with grey). With this structure it is possible to directly implement the warping filter without unwarping the coefficients first [6][5][7].

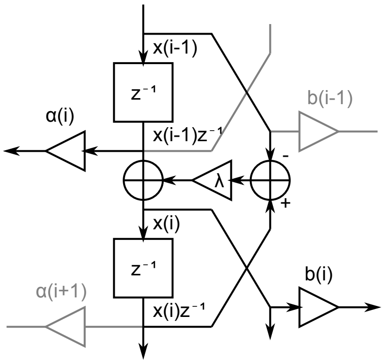


Figure 1: Implementable WIIR stage

III. Simulating Fixed-Point Filters

In many audio applications the lower frequency poles and zeros of a filter are more important. Based on the auditory scales the lower frequency ranges should have higher resolutions. Therefore designing a filter on a linear frequency scale ends up with higher filter orders compared to those of filters designed on logarithmic frequency scale, supposing the same subjective results.

Another problem is the position of the poles and zeros. If the filter poles and zeros are crowded in the lower frequency range, the fixed-point implementation of that filter will suffer from greater quantisation errors. Warping the frequency scale helps to separate these poles and zeros making the quantisation error have less effect on the actual filter implementation.

A. *Arithmetic Logical Unit*

To investigate the behaviour of fixed-point IIR and WIIR filters from quantisation error point of view, a simulation environment was created in MATLAB based on the most commonly used DSP architectures. In order to emulate the DSP core, a simple arithmetic logical unit (ALU) was created as seen in Figure 2. This structure consists of three stages. From top to down the first one represents the memory where the ALU can get the parameters from. This is the input-output interface as well, to pass variables from or to other calculations. The second stage represents the registers and the register bus, and the third stage represents the accumulator register and its bus. There are several ways to transfer data from one section to another, all of them are marked with a circle and a corresponding quantisation function.

B. *Quantisation properties*

In the current framework quantisation can be applied as:

- parameter quantisation: for filter coefficients.
- register quantisation: for the register bus.
- accumulator quantisation: for the accumulator bus.

In all of these three quantisations it is possible to set bit depth for the range $[-1; 1[$ and the saturation limit. For example having a 16 bit depth set for the range $[-1; 1[$ and the saturation set to 17 bit, the $[-1; 1[$ will be divided into 2^{16} steps, although saturation is only applied by reaching the $[-2; 2[$ range limits. In the simulation toward zero rounding is used as in most DSP implementations to avoid limit cycles.

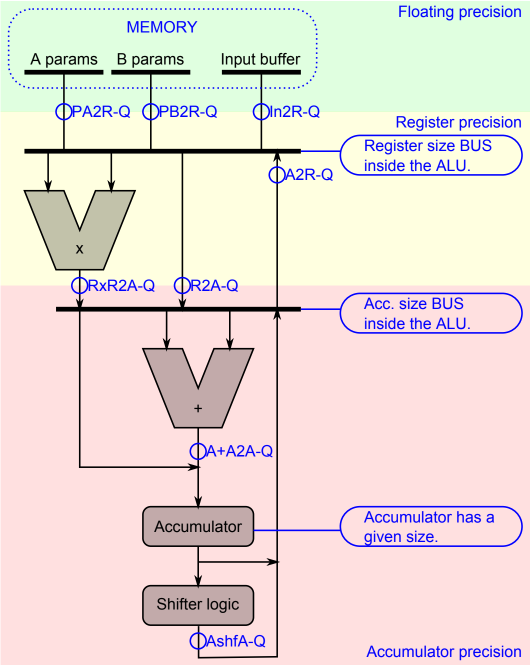


Figure 2: The ALU structure

C. Macro assembly language

On the top of the ALU simulation a pseudo macro assembly language was developed containing the basic instructions: addition, multiplication and multiply and accumulate (MAC). This pseudo language wraps the quantisation and uses the right quantisation property-set emulating the data transfer between the sections of the ALU as well. This ensures that the filters implemented with this macro assembly language will use the correct quantisation functions.

D. Filter implementation

The simulation contains all well known IIR filter implementations using the above mentioned macro assembly language. This means IIR direct form 1, IIR direct form 1 transposed, IIR direct form 2, IIR direct form 2 transposed, and the WIIR filter. There are additional functions to split up long IIR filters to serial second-order sections and parallel second-order sections, and there are functions to unwarp WIIR second-order sections to normal IIR second order sections. And of course all IIR filters can be used as FIR filters if needed.

E. Results

The first attempt was to compare WIIR filter and the unwarped serial second-order section IIR filters. The flowchart is shown in Figure 3. As seen there, first the target transfer function is warped using (1). This is done by summing up several transfer functions created with the all-pass filter. Then using Prony's method an IIR filter is designed, returning the a and b coefficients for the filter. Based on these parameters and the equations (3)-(5) the α parameters can be calculated for the implementable WIIR structure. On the other hand a and b coefficients can be split into serial second-order sections and directly unwarped to normal second-order IIR filters.

The results of the simulation with 16 bit register bus size and 32 bit accumulator bus size without any saturation and with filter order $N = 50$ can be seen in Figure 4. Comparing normally designed IIR filter with the WIIR filter and the unwarped serial second-order stage IIR filter performance. The non-quantised filters' magnitude responses can be seen in the upper diagram. The WIIR fits fully the unwarped SOS IIR so does the IIR the SOS IIR.

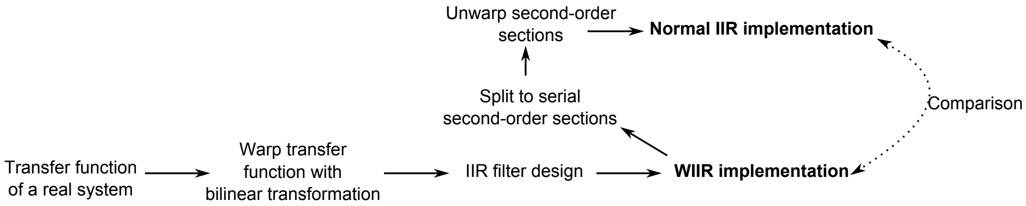


Figure 3: The flowchart of the simulation

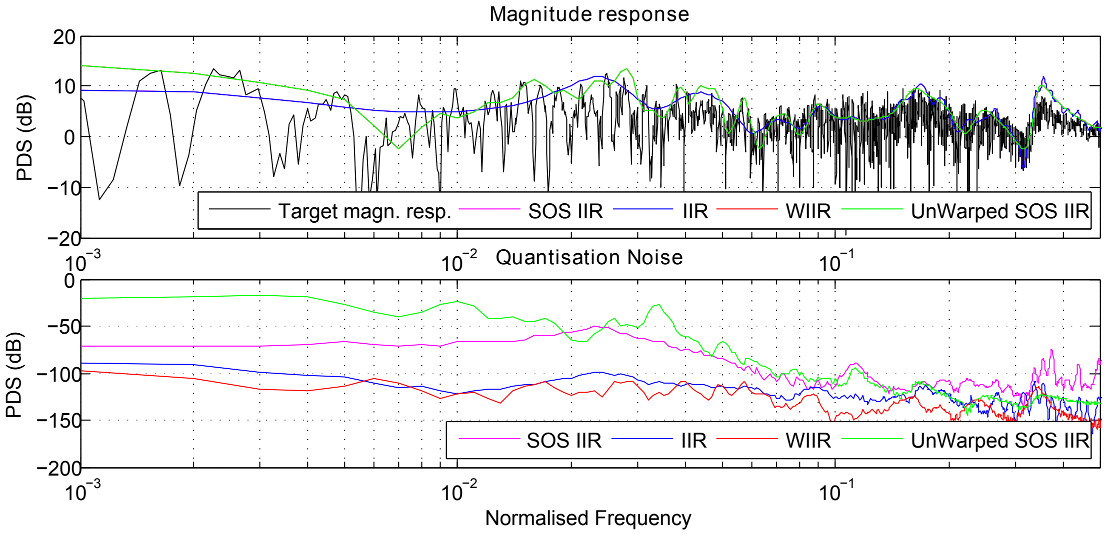


Figure 4: Simulation results

IV. Conclusion

As shown in this report a MATLAB simulation framework was created and first comparisons were made to investigate the quantisation error behaviours of several IIR filter structures. As seen in Figure 4 the WIIR has the best performance. It fits with higher resolution on an auditory scale compared to normal IIR, and has the same amount of quantisation noise. The unwarped SOS IIR has increased quantisation noise, but a higher resolution in the low frequency range than the normal IIR.

In the near future all the filter types will be compared to each other in different scenarios and the noise model of the WIIR will be calculated validating so the simulation framework. Further investigations will be done on the possibilities of warping by using higher order all-pass filters.

References

[1] W. Peng, W. Ser, and M. Zhang, “Multiband warped filter equalizer design for loudspeaker systems,” in *Acoustics, Speech and Signal Processing, IEEE International Conference*, pp. 913–916, Istanbul, June 5–9 2000.

[2] A. Harma, M. Karjalainen, S. L., V. Valimaki, U. Laine, and J. Huopaniemi, “Frequency-warped signal processing for audio applications,” *J. Audio Eng. Soc.*, 48(11):1011–1029, Nov. 2000.

[3] M. Karjalainen, A. Harma, U. Laine, and J. Huopaniemi, “Warped filters and their audio applications,” in *Applications of Signal Processing to Audio and Acoustics, IEEE ASSP Workshop*, New Paltz, NY, Oct.19–21 1997.

[4] A. Harma, “Implementation of recursive filters having delay free loops,” in *Acoustics, Speech and Signal Processing, IEEE International Conference*, pp. 1261–1264, Seattle, WA, May12–15 1998.

[5] M. Karjalainen, A. Harma, and U. Laine, “Realizable warped IIR filters and their properties,” in *Acoustics, Speech, and Signal Processing, IEEE International Conference*, pp. 2205–2208, Munich, Apr.21–24 1997.

[6] M. Karjalainen, A. Harma, and U. Laine, “Realizable wareped IIR filter structures,” in *IEEE Nordic Signal Processing Symposium*, pp. 483–486, Espoo, Finland, Sept.24–27 1996.

[7] A. Harma, “Implementation of frequency-warped recursive filters,” *Elsevier Signal Processing*, 80:543–548, 2000.

VERSION CONTROL IN FPGA BASED MULTIPROCESSOR SYSTEM DEVELOPMENT

István SZIKRA
Advisor: Béla FEHÉR

I. Introduction

Because Xilinx does not have integrated version control in their products, when we started researching and testing multiprocessor systems we found that it is rather hard to keep track of the files that the Xilinx Integrated Development Environment produces. It was important to trace what files were needed to reproduce the designed system exactly, and what files can be considered useful, that contain data and reports about the designed system that later can be extracted by automated tools to create progress reports.

Integrated Development Environments (IDEs) are a blessing, but they are also a curse. They make it easy to develop simple projects (or projects that seem simple in the IDE), but they can get difficult when we are working on complex systems. They also hide the inner operations of the development process, the tools they call and the files they change. Changing some properties on a complex system in the GUI takes what sometimes seems forever. The same changes could be made more quickly in text editors, and then running the necessary batch processes. But for this, we have to know what files should be changed and what files should be generated. Unfortunately, Xilinx is not that forthcoming about a complete list of files used by their IDE, and they change some of them from version to version. A few month ago the latest version of the Xilinx ISE was 14.2, and by the time of writing this paper, they already have put out two new versions 14.3 and 14.4. Because Xilinx has thousands of pages of manuals just for the development environment, not to mention the other thousands of pages for the devices, it is easy to get lost in them. Google does not help much either. There are a lot of outdated or incomplete results about the file types the Xilinx IDE uses.

II. Version Control

Version Control is designed to keep track of files. So it is a good tool to find out what is happening with files in a directory. Other tools like file monitors (for example FileMon in the Sysinternals Suite) can be used to find out what process is accessing certain files, but we are more interested with the end result.

If you write software code, there are a few well documented types of files you work on depending on the programming language, which is not the case with Xilinx and Eclipse CDT tools. Using external tools adds more file types that need to be tracked for configuration changes. Creating systems that use both hardware and software development tools and sources complicates things further. When we create even a simple soft-core multiprocessor design we end up with thousands of files consuming hundreds of megabyte of disk space. This gets unmanageable rather quickly as we are creating more complex systems to compare them against each other. We have to sort out which files are important. This is where reading manuals, searching the Internet and/or using a folder monitoring tool like version control is necessary.

So why use Version Control? Mainly because it keeps track of your files for you, instead of you, once you told it what to monitor. So after changing things in a complex system you will be able to go back to any previous state. You can relax and concentrate on the development, trying out new ideas. You do not have to worry if you or the IDE changes things. Thanks to version control, you can go back to any committed version if you need to. Working without version control is like working without a net in circus. Does it really worth the risk? There are many reasons why you should use version control, and there are a lot of things it can do for you. One example is to use the version control commit log as a change log, or create a progress report. If we are doing it right it can also tell us not just what we did, but why we did it.

One slight problem I mentioned is figuring out what files should be kept under version control (or backed up for that matter). Of course it is not designed to solve this problem, but it can help finding out which files to keep track of. Just commit and trace everything. Look for the modified lines in files when you change some configuration in the GUI. Make notes of these (in the version control commit log). Next time you can do it from script without the GUI and automate reconfiguration of many processor cores at once. This way we can learn a lot about even an unfamiliar IDE. Keep looking for modifications when you are not actually changing any source file or configuration. These are the files, which should be ignored. It will keep our projects small and free of clutter.

III. Results

Multi core test project sizes ranged from 40MB (using only hard core processor) to 350MB with 1000 files to more than 2000 files.

I found that the files you should commit and keep under version control to make the system reproducible is as follows.

- for simple ISE projects: *.v, *.vhd, *.ucf, *.tcl and *.xise files;
- for EDK (XPS) projects: *.xmp, *.mhs, data/* (*.ucf, *.xml), pcores/*, etc/*, hdl/* files and folders;
- in SDK for the hardware platform: system.xml, .project, system.bit files in the *_hw_platform folder;
- SDK Board Support Package files: .cproject, .project, .sdkproject, libgen.options, Makefile, system.mss;
- for every SDK application: .cproject, .project, src/* (*.c, *.h, *.ld) files and folders;
- for PlanAhead projects: *.ppr file and the *.sdk/ like the SDK, the *.srcs/ like the EDK (XPS).

Some of the html report files can also be kept for later to extract progress information.

IV. Conclusion

Version Control System is a great and universal tool not just tracing modification of known files, but also for identifying relevant files, resources in a project. It can be used to explore file and folder structures of complex development environments such as the Xilinx has by tracing each and every file during a development process. The intended use is to highlight the changes we are interested in, but we can use it to show the irrelevant changes that occur during the development of soft-core based designs that we can ignore.

References

[1] Scott Chacon, *Pro Git (Expert's Voice in Software Development)*, Apress, August 27, 2009, URL: http://git-scm.com/book.

[2] Bryan O'Sullivan, *Mercurial: The Definitive Guide*, URL: http://hgbook.red-bean.com/read/.

[3] Xilinx, *Using SDK with Revision Control Software*, URL: http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_3/SDK_Doc/reference/sdk_u_cvs.htm

[4] Xilinx, *Xilinx SDK Command Line Flows*, URL:http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_3/SDK_Doc/reference/sdk_u_commandline.htm.

3D MEMS SENSOR ON TRUCK TYRES FOR SAFETY CRITICAL APPLICATIONS

András JÁNOSI
Advisor: Tamás DABÓCZI

I. Introduction

This review focuses on automotive embedded systems, especially on trucks. There are plenty of safety critical electronic systems on trucks, such as Slip Control Systems, Emergency Braking System (EBS), Electronic Stability Programme (ESP), Adaptive Cruise Control (ACC) to name a few. Slip control systems (like ABS), ESP and EBS use slip and friction values. If some of their sensors (e.g. accelerometers) fail, the slip and friction values have to be estimated somehow, consequently the need arises to substitute the corresponding measurements by the estimations. Inverse method is a good choice in these cases, because it is able to estimate sensor values from other measurements if their relation or model are known. This is called analytic redundancy. Nowadays, there is more and more attention placed on researching the intelligent tyre, which is able to fulfill this task for slip control and stability systems.

In the framework of the ENIAC project, the Institute for Technical Physics and Materials Science (MFA) from the Research Centre for Natural Sciences (part of Hungarian Academy of Sciences) reserched and developed a special MEMS based 3D force sensor, which can be integrated into vehicle tyres to measure forces. The future research will use this sensor to substitute or supplement some instruments, which are already deployed on trucks.

In the following sections we will present the latest researches on intelligent tyres, then will introduce the sensor designed in the ENIAC project followed by an outlook of the possible future research.

II. Summary on intelligent tyres

There are two types of approaches for estimating the friction coefficient: indirect and direct methods. The indirect method does not monitor tire deformation while the direct method does. In most indirect methods, the friction coefficient is determined based on the vehicle velocity, wheel angular speed, and normal and tractive forces applied to the tire, axis and wheel. Since the relationships among tire parameters are heavily nonlinear and complex, analytic relations are difficult to acquire [1]. Therefore, algorithms such as a fuzzy logic controller [2] or Kalman filter [3] are used to estimate tire parameters. Yi et al. [4] used the wheel slip, vehicle velocity, and normal load on the tire to determine the friction coefficient and develop a control scheme for emergency braking maneuvers. Miyasaki et al. [5] measured the four-axis direction force by attaching strain gages to improve the ABS efficiency. Ohori et al. [6] measured the strains applied to the wheel to estimate the six force components in the tire.

Direct sensor allows a precise measurement of tire deformation or strain. Surface acoustic wave (SAW) sensors have been proposed for monitoring the deformation during road contact [7]. SAW devices use metallic interdigital transducers arranged on the surface of a piezoelectric substrate. Palmer et al. [8] demonstrated the embedment of fiber optic sensors in an automobile tire for monitoring tire strain and captured and measured the onset of skid. Tjiu et al. [9] used microelectromechanical system (MEMS) sensors, including a pressure sensor, accelerometer and temperature sensor for a tire condition monitoring system. Without attaching sensors, Matsuzaki et al. [10] presented a self-sensing method using the tire structure itself as a parallel circuit of a capacitor and resistor.

III. Introduction of the sensor, developed within the framework of ENIAC project

In the MEMS laboratory of MFA in collaboration with WESZTA-T Ltd, from 2009 to 2011, a 3D force sensor has been developed, that can be used in an automotive tire for the measurement of longitudinal and lateral forces [11] [12]. The silicon single cristal force sensor has been encapsulated into rubber patches to be able to fix it on the sidewall of tires. Several succesful tests have been performed with the sensor in laboratory conditions on a test benchs and on vehicles equipped with the sensor. This icludes test on cars and trucks too.

The positive results of the tests makes the sensor convenient to apply as a redundant sensor for the earlier mentioned safety critical applications on trucks.

IV. Future researches

With the given sensor and the corresponding measurement system the following steps will be carried out in the close future:

- Explore the potencial sensors on trucks that can be substituted or complemented by the 3D MEMS sensor.
- Test measurement on trucks, with reference measurements, eg. with accelerometers. This test series will be performed with the support of Knorr-Bremse Brake Systems Ltd.

V. Conclusion

Until the present state of this research, the goals were to explore the latest similar researches, to get knowledge about the sensor developed by MFA, finally set the potential resarch targets.

Acknowledgment

Special thanks to Gábor Battistig from Research Institute for Technical Physics and Materials Science and Richárd Végvári from WESZTA-T Ltd for their help in studying the sensor.

References

[1] C. Lee, K. Hedrick, and K. Yi, "Real-time slip-based estimation of maximum tire-road friction coefficient," *IEEE/ASME Trans. Mechatron.*, 9:454–458, 2004.

[2] G. F. Mauer, "A fuzzy logic controller for an abs braking system.," *IEEE Trans. Fuzzy Syst.*, 3:381–388, 1995.

[3] F. Gustafsson, "Monitoring tire-road friction using the wheel slip.," *IEEE Contr. Sys. Mag.*, 18:42–49, 1998.

[4] J. Yi, L. Alvarez, and R. Horowitz, "Adaptive emergency braking control with underestimation of friction coefficient.," *IEEE Trans. Contr. Sys. T.*, 10:381–392, 2002.

[5] N. Miyasaki, M. Fukumoto, Y. Sogo, and H. Tsukinoki, "Antilock brake system (m-abs) based on the friction coefficient between the wheel and the road surface.," *SAE Tech. Papers*, 1990.

[6] M. Ohori, T. Ishizuka, T. Fujita, N. Masaki, and Y. Suizu, "Fundamental study of smart tire system.," in *Proceedings of 2006 IEEE Intelligent Transportation Systems Conference*, pp. 1519–1524, Toronto, Canada, 2006.

[7] A. Pohl, R. Steindl, and L. Reindl, "The intelligent tire utilizing passive saw sensors - measurement of tire friction.," *IEEE Trans. Instrum. Meas.*, 48:1041–1046, 1999.

[8] M. Palmer, C. Boyd, J. McManus, and S. Meller, "Wireless smart tires for road friction measurement and self state determination.," in *43rd AIAA/ASME/ASCE/AHS Structures, Structural Dynamics, and Materials Conference*, Denver, Colorado, USA, 2002.

[9] W. Tjiu, A. Ahanchian, and B. Majlis, "Development of tire condition monitoring system (tcms) based on mems sensors.," in *Proceedings of IEEE Intl. Conf. on Semiconductor Electronics*, Kuala Lumpur, Malaysia, 2004.

[10] R. Matsuzaki and A. Todoroki, "Wireless strain monitoring of tires using electrical capacitance changes with an oscillating circuit.," *Sens. Actuat. A*, 119:323–331, 2005.

[11] I. Bársony, "Professional report of eniac-se2a mfa 2011," pp. 12–14, 2011.

[12] G. Battistig, T. Weidisch, T. Retkes, M. Ádám, I. Bársony, and T. Mohácsy, "Thermal behaviour of three dimensional single crystalline force sensors," *21st Micromechanics and Micro systems Europe Workshop*, 2010.

SCHEDULING IN WIRELESS SENSOR NETWORKS (SUMMARY OF PHD WORK IN 2012)

Péter GYÖRKE
Advisor: Béla PATAKI

I. Introduction

In wireless sensor networks (WSNs) the power supply for sensor nodes is critical. The main goal is to extend the life cycle of a sensor node. There are two non-contradictory approaches to achieve this goal: 1) expand the power source: larger battery capacity or energy harvesting techniques; 2) reduce the energy consumption of the sensor nodes. The first approach is rather technological: development of new types of batteries or solar cells is out of the scope of our research. The energy consumption can be divided in two parts: 1) the energy consumption of the hardware components [1] (also a technological problem to reduce it); 2) the energy consumption resulted by the operation modes, which is depend on the software of the nodes. The operation modes can be the following: 1) measure; 2) communicate; 3) sleep. Each mode has different energy consumption. In some cases the measurement has similar or less consumption compared to the consumption of the sleep mode, usually the communication is the critical phase.

Table 1: Energy consumptions of a sensor node in different modes. Case 1 is when the measurement has a significant consumption; in case 2 this is negligible.

Operating mode	Case 1	Case 2
Measure	0.2 mA	0.002 mA
Sleep	0.005 mA	0.005 mA
Communicate	10 mA	10 mA

According to Table 1. in case 1 the measurements (and the communication) are to be scheduled and in case 2 the communication needs to be scheduled in order to achieve lower energy consumption. It should be emphasized that the consumption of communication is indifferent from the direction, so the reception of data also has a high cost. For that reason, when the sensors are in sleep mode, communication towards the sensors is not possible. The reconfiguration of the sensors (setting of sampling rates) is only reasonable in the acknowledgement message after a successful transmission from the sensor (see Fig 1. where “A” means: Motion event happened or a scheduled transmission by the sensor. “B” means: Coordinator processes the new data and calculate the next wake-up time for this sensor and send acknowledgement. “C” means: The sensor processes the acknowledgement and goes into sleep mode for the given time.).

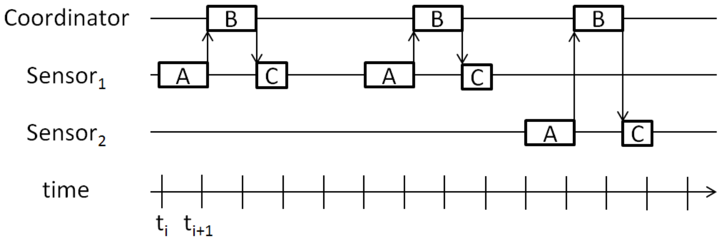


Figure 1: Sequence diagram of the communication between the coordinator and two motion sensors.

II. Measurement scheduling

The problem in brief: how to select the sampling rates of the measurements to reduce energy consumption, but maintain or control the measurement precision of the whole network.

A Hidden Markov Model (HMM) was used to model the problem. To create the model, the states, state transitions, a priori probabilities, sensor models have to be defined. A case study has been presented using this method to schedule measurements of a motion sensor network. The purpose of the sensor network is to track a person's movement in indoor environment (e.g.: an apartment). The HMM models the motion of the person, the problem is transformed into a finite state one, because spatial quantization is applied. The state vector represents the probability of positions of the person in the apartment. The state transition matrix defines the probabilities of moving from one position to another.

The notations are the following for this actual example:

- k is the (discrete) time,
- $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{46}\}$ is the set of possible states (in our simulation 46 positions are used),
- $S = \{s_1, s_2, ..., s_{20}\}$ is the set of sensors used (in our simulation 20 sensors are used),
- $O_i = \{0,1\}$ is the finite set of symbols the i^{th} sensor can provide

The state transition matrix (**A**) is defined:

$$\mathbf{A} = \begin{bmatrix} P(x_k^1|x_{k-1}^1) & P(x_k^2|x_{k-1}^1) & \cdots & P(x_k^{46}|x_{k-1}^1) \\ P(x_k^1|x_{k-1}^2) & P(x_k^2|x_{k-1}^2) & \cdots & P(x_k^{46}|x_{k-1}^2) \\ \vdots & \cdots & & \ddots \ \vdots \\ P(x_k^1|x_{k-1}^{46}) & P(x_k^2|x_{k-1}^{46}) & \cdots & P(x_k^{46}|x_{k-1}^{46}) \end{bmatrix} \tag{1}$$

We used a 10-day simulation and from first 5 days we accumulated the state transitions, so the A matrix contains the most possible routes for the target. The state vector (**X**) is the following:

$$\mathbf{X}_k = \begin{bmatrix} x_k^1 = P(pos = 0|y_{1:k}) \\ x_k^2 = P(pos = 1|y_{1:k}) \\ x_k^3 = P(pos = 2|y_{1:k}) \\ \vdots \\ x_k^{46} = P(pos = 45\,|y_{1:k}) \end{bmatrix} \tag{2}$$

$$|\mathbf{X}_k| = 1 \tag{3}$$

where k is the time index, $P(pos = i\,|y_{1:k})$ means the probability of the target is in "i" coded position if the measurements were $y_{1:k}$. We have to calculate the probability of the next state in two cases. If there are new measurements:

$$\mathbf{X}_{k+1} = \alpha \cdot diag(\mathbf{E}_k)\mathbf{A}^{\mathbf{T}}\boldsymbol{X}_k \tag{4}$$

where the probability of the new measured information is represented by $\mathbf{E}_k$, this is calculated from $\mathbf{Y}_k$ and from the sensor model. If there are no new measurements, we can predict one step according to the state transition matrix:

$$\mathbf{X}_{k+1} = \alpha \cdot \mathbf{A}^{\mathbf{T}}\mathbf{X}_k \tag{5}$$

The output of the HMM is a probability vector, representing the likelihood of the different positions of the observed person.

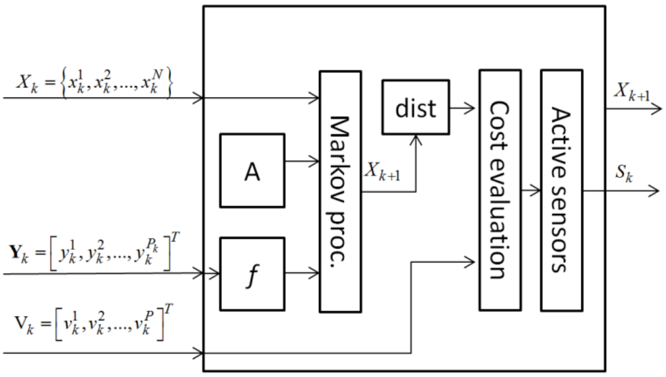


Figure 2: The block diagram of the system proposed for the actual application (V_k is the storage voltages of the sensors operated at k, S_k is the vector of the sensors reconfigured in the time *k*).

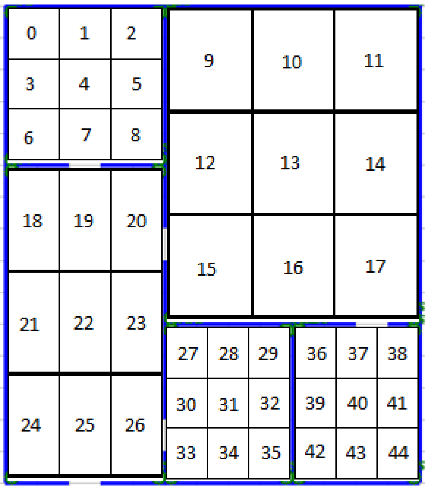


Figure 3: The observed appartment is divided to these subsections. Each room is divided to 9 subsections and the 45th outside subsection is not marked on the figure.

Sensor selection

The sensor's sampling rates can be configured in two settings: normal rate, increased rate. Based on the output of the HMM and the actual state of the sensors energy levels, some sensor will be configured to increased rate, others will be kept on normal rate. The selection is based on a cost function:

$$C(i,j,k) = \mathrm{rel}(i,j) + \frac{\alpha}{\mathrm{voltage}(i,k)} \tag{6}$$

where *i* is the sensor index, *j* is the index of the most possible state, α is a tuning parameter, it defines the weight of the energy level in the cost function. The distance function defines the relevance of the sensor to the most probable actual position of the target. The distance is calculated using the state transition matrix (**A**): the shortest possible path from the actual most probable state (position) to the state closest to the sensor is searched for. The rel(*i*,*j*) is a relevance function which results in a value inversely proportional to the relevance of the sensor to the estimated state. We can calculate that how many state transitions are needed from the actual most possible state to get in a state closest to the sensor. Optionally we can weight this number with the probabilities of these state transitions. We can substitute this calculation with a look-up-table, because it is independent from the actual time *k*. The sensor's energy level is represented by the voltage of its storage capacitor. It is a good estimate, due to the linear voltage curve of a capacitor when continuous current is drawn from it (the model is proven by measurements in [1][2]). We will put on the increased rate sensor list the first L sensors with the lowest cost. L is a parameter of the model. Optionally L can be substituted with a cost limit parameter. We calculate the costs to create an active sensor list in every *k* moment, but the particular sensors will be reconfigured only when a transmission exchange is performed with the coordinator.

The proposed method was verified by simulation, further details in [3]. The results of a 10 day simulation are presented in Fig. 4. In the first 5 days only the **A** transition matrix was learned. The error is calculated from the distance of the simulated position and the center points of different subsections and it is averaged over the time. As expected the fixed measurement interval method produced the most accurate estimation in position, but it also has the highest energy demand. The “Simple” method maintained an active room where the last movement has taken place. The sensors in the active room are set to increased rate, others to normal rate. The time-of-day (ToD) method is worth mentioning from the basic methods. It produced remarkably better results than the other basic methods, but it is due to the used a priori information (night time and day time). The HMM method produced even better results (especially in consumption), and it did not utilized a priori information, the state transition matrix generated on-line.

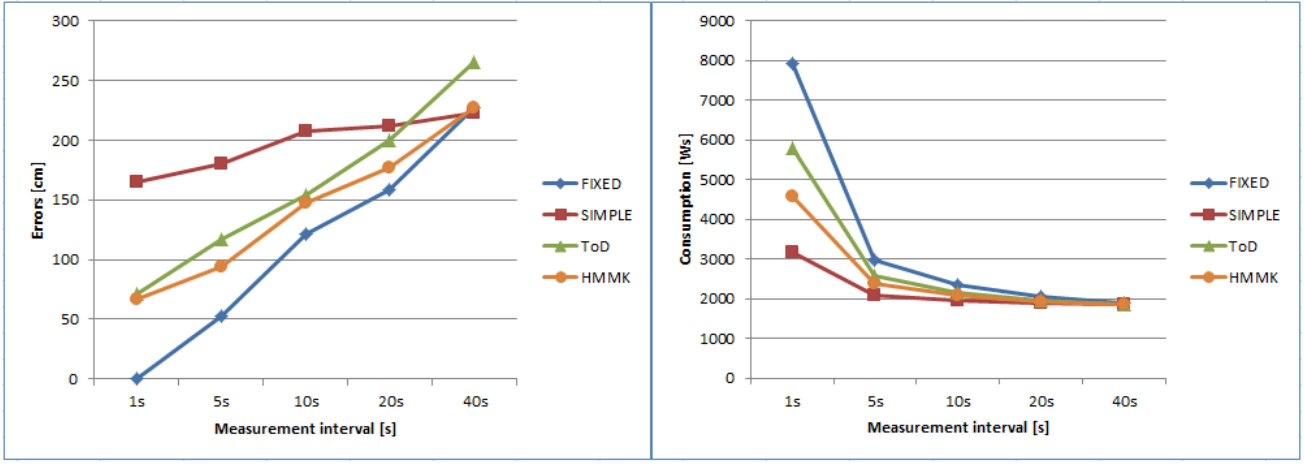


Figure 4: The resulted errors (left) and energy consumptions (right) od the simulation for each scheduling methods.

III. Further work

The case where the measurement cost is negligible should be further investigated. The transmission of the measurement could depend on the measured data and previous measurements of the other sensors. There was a strict limitation in the previously presented system: the sensors can be reconfigured after a data transmission only. If we can assume a low power radio system which can notify or reconfigure the sensors, various new methods can be elaborated. To prove these methods a real life test should be performed.

References

[1] P. Györke, B. Pataki, *Power supply design for solar powered harvesting-aware sensors*, Pollack Periodica, 7/3, pp. 45-54, 2012.

[2] P. Györke, B. Pataki, *Application of energy-harvesting in wireless sensor networks using predictive scheduling*, IEEE I2MTC Conference 2012. Graz, Ausztria, 2012.05.13-2012.05.16., pp. 582-587.

[3] P. Györke, B. Pataki, *Energy aware measurement scheduling in WSNs used in AAL applications*, IEEE Transactions on Instrumentation and Measurement, 2012. Accepted.

QDF^2: A Kernel-based Data Fusion Framework for Drug Repositioning

Bence BOLGÁR
Advisor: Péter ANTAL

I. Introduction

During the past decade, the pharmaceutical industry has experienced a continuous decrease in the number of new molecular entities (NMEs) per year, whereas the R&D spending has been constantly rising. In response to these disturbing trends, also known as the "pharma innovation gap", a range of new drug discovery strategies have emerged. Drug repositioning, the innovative application of a known drug in a new indication, has been considered as a particularly promising approach. In this work, we present a data fusion framework which aims to support drug repositioning by utilizing multiple heterogeneous information sources [1].

II. Kernel-based data fusion

Using data fusion to support *in silico* drug discovery is not a new concept. The earliest methods exploited the similar property principle (SPP), which states that similar compounds tend to have similar properties. Traditional similarity-based approaches, such as rank-based data fusion (RBDF) focused on finding drug candidates by using positive examples and combining multiple similarities. However, a number of questions remained open, including:

- Multiple entities (one-to-many similarities)
- Automatic weighting of the information sources (*e.g.* query-driven adaptive fusion)
- Statistical guarantees

Kernel-based data fusion (KBDF), also known as Multiple Kernel Learning addresses all of these questions and has been successfully applied in gene prioritization [2]. The idea is to formulate the problem as a one-class SVM classification problem, but instead of a classification step, the samples are prioritized on the basis of their distance to the origin. In this work, we adapted the formulation of Vishwanathan et al. [3] to this one-class prioritization task:

$$\begin{aligned} \min_{f_k,\rho,\boldsymbol{\xi},\boldsymbol{d}} \quad & \frac{1}{2}\sum_k \frac{\|f_k\|^2_{\mathcal{H}_k}}{d_k} - \rho + \frac{1}{\nu l}\sum_i \xi_i + \frac{\lambda}{2}\left(\sum_k d_k^p\right)^{\frac{2}{p}} \\ s.t. \quad & \sum_k f_k(\boldsymbol{x_i}) \geq \rho - \xi_i, \\ & \boldsymbol{\xi} \geq 0, \quad \boldsymbol{d} \geq 0, \quad i = 1, 2, \ldots, l \end{aligned}$$

This corresponds to the one-class SVM primal with the additional capability to handle multiple representations weighted by $\boldsymbol{d}$. The last term stands for the regularization of the kernel weights and we chose it to be the squared $L2$-norm regularizer to promote density across the kernels. The dual is

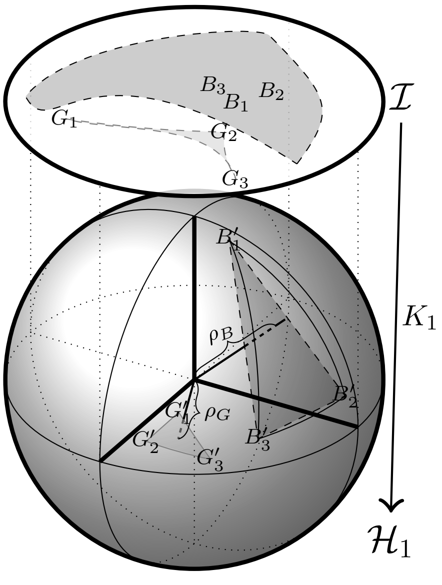
$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\frac{1}{8\lambda}\sum_k \left(\boldsymbol{\alpha}^T K_k \boldsymbol{\alpha}\right)^2 \\ s.t. \quad & 0 \leq \alpha_i \leq 1, \quad \mathbf{1}^T\boldsymbol{\alpha} = \nu l, \end{aligned}$$

which can be solved very efficiently by the SMO algorithm and the kernel weights can be recovered from the solution. The samples can be prioritized by projecting onto the normal of the hyperplane and computing the normalized distance to the origin:

$$score(\boldsymbol{x}) = \frac{\sum_i \alpha_i \sum_k d_k K_k(\boldsymbol{x_i}, \boldsymbol{x})}{\sqrt{\sum_k d_k \boldsymbol{\alpha}^T K_k \boldsymbol{\alpha}}}.$$

Furthermore, we utilized invariant kernels which satisfy $K_k(\boldsymbol{x_i}, \boldsymbol{x_i}) = 1$. Since in this setting the samples in the Reproducing Kernel Hilbert Spaces $\mathcal{H}_k$ lie on the surface of a hypersphere, it follows that learning the optimal separating hyperplane is equivalent to finding a kernel weighting which minimizes, loosely speaking, the "volume" of the training set in the combined RKHS $\mathcal{H} = \bigoplus_k \{f | f \in \mathcal{H}_k : \frac{\|f\|_{\mathcal{H}_k}}{d_k} < \infty\}$.

The figure on the right shows a RKHS $\mathcal{H}_1$ in which the query G has a "compact" kernel representation *i.e.* the invariant kernel K_1 captures G fairly well (equivalently, the margin ρ_G is close to 1). This implies that K_1 would gain a high weight in this scenario. However, if applied to the query B, the same kernel would gain a much lower weight. Since the kernel weighting depends on the training set (intuitively, KBDF actively exploits the "structure" of the question), we can see that the KBDF method avoids a major drawback of RBDF, which resorts to global fusion schemes.



$\rho_B < 1, \quad \rho_G \approx 1$

III. Results and future work

The main idea is to define the training set as a group of drugs which are approved in a certain indication, then to hunt for similar drugs (very much like a "biomedical Google"); this can be formulated as a prioritization problem with some appropriate kernels. To this end, we utilized various pharmacological information sources to construct the kernel representations of the drugs.

Our systematic evaluation showed that KBDF outperformed the traditional RBDF approach by predicting ATC-class memberships with a significantly better performance in a 100-fold 70-30 cross-validation setting, using multiple performance measures. We also performed a case-by-case validation using expert knowledge and found several candidates for repositioning.

Current developments include an OpenCL-based implementation for GPU clusters, a general prioritizer software pack with GUI and various mathematical extensions.

Acknowledgements

The author thanks Ádám Arany, Dr. Balázs Balogh, Dr. Péter Antal, Prof. Dr. Péter Mátyus for their contributions. The project was funded by NKTH TECH 08-A1/2-2008-0120.

References

[1] A. Arany, B. Bolgar, B. Balogh, P. Antal, and P. Matyus, "Multi-aspect Candidates for Repositioning: Data Fusion Methods Using Heterogeneous Information Sources," *Curr. Med. Chem.*, Nov 2012.

[2] T. De Bie, L. C. Tranchevent, L. M. van Oeffelen, and Y. Moreau, "Kernel-based data fusion for gene prioritization," *Bioinformatics*, 23:i125–132, Jul 2007.

[3] S. V. N. Vishwanathan, Z. Sun, N. Theera-Ampornpunt, and M. Varma, "Multiple kernel learning and the SMO algorithm," December 2010.

A STATISTICAL APPROACH TOWARDS RIBCAGE SEGMENTATION (SUMMARY OF PHD WORK IN 2012)

Áron HORVÁTH
Advisor: Gábor HORVÁTH

I. Introduction

Ribcage segmentation is an essential step for bone shadow removal on chest radiographs. Ribs are surrounding the thorax, therefore most ribs have two shadows on an image. In this paper I will focus on finding the posterior part of the ribs, but some techniques can be applied to the anterior case as well.

There are several approaches towards ribcage segmentation. One of these is to analyze vertical intensity profiles of the images. This method was introduced by Souza [1], who synchronized these profiles by shifting. Then summed them into a single one-dimensional profile, on which he applied a t-test to enhance the rib borders. After local minima and maxima selection he got approximal rib borders. Here, I would like to present a new procedure which can be regarded as a revision of Souza's work. I will focus on the part which is related to Souza's work, the overview of my full solution can be found in [2]

II. Description of the method

One problem with the aforementioned method is that simple shift can not put these profiles into exact synch, because each rib has a slightly different slope. To overcome this I fit a smooth slopefield to the image gradients and synchronize the vertical sections based on it. The base section of the synchronization will be the yellow dotted curve of fig. 1d, and the intensities are averaged over the red curves.

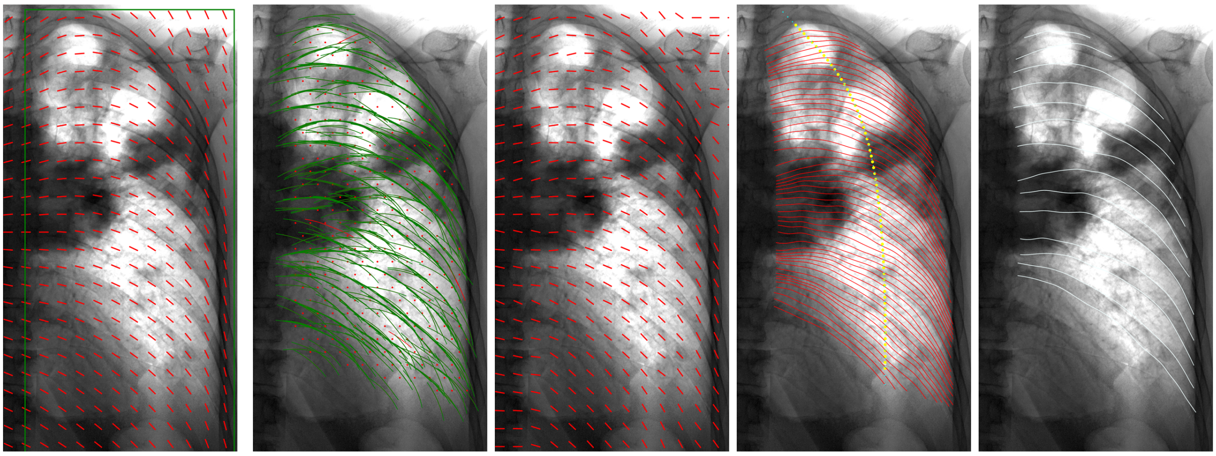
The averaged intensity profile is depicted by the blue curve of fig. 2a. Instead of t-tests I applied a box filter (fig 2b), differentiation (fig 2c) and normalization, which encompasses a highpass filter and sliding RMS compensation (fig. 2d).

A simple peak-picking routine is not sufficient to locate the rib edges, because usually there are more extrema than rib borders, only images of young and healthy individuals lead to clear-cut cases.

To address this problem a dynamic programming approach is introduced. The mechanics of the search are as follows. The algorithm passes through the signal from the left to the right. Some points on the left side are possible upper borders. If intercostal spaces and rib thicknesses are supposed to change gradually, picking one point as upper border determines the possible position range for the corresponding lower border. Choosing a lower counterpart determines the permitted range for the next rib thickness. The third step, where the upper border of the second rib is selected, constrains the possible spacing range for the following ribs, and it goes like this. These ranges were defined by analyzing a set of manually segmented ribcages.

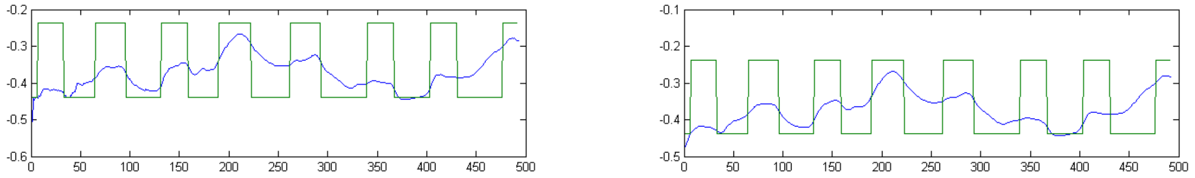
4 different statistics were created (fig. 3). All of them describe a dependence on the vertical position inside the lungfield: rib thickness, spacing between ribs, consecutive rib thickness ratio, consecutive spacing ratio.

The vertical position is measured from the apex of the lungfield along the midcurve (yellow dotted curve in fig 1d). This curve is not always perpendicular to the ribs, thus the distance measured on it between two rib borders can be a slightly more than the thickness of the rib. This is compensated by measuring distances based on the slopefield, accounting only the part which is perpendicular to it. The one dimensional signal is created utilizing this measure, too.

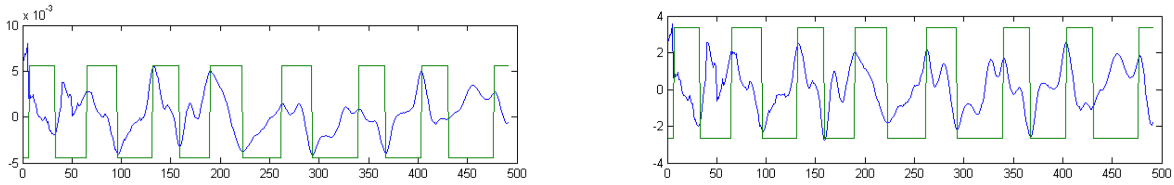


(a) Average slope field (b) Fitted arcs (c) Fitted slope field (d) Possible curves (e) Selected curves

Figure 1: Overview of posteriar ribcage segmentation. Arcs are fitted to the image to finetune an average slopefield, then some of the curves generated from this slopefield are selected.

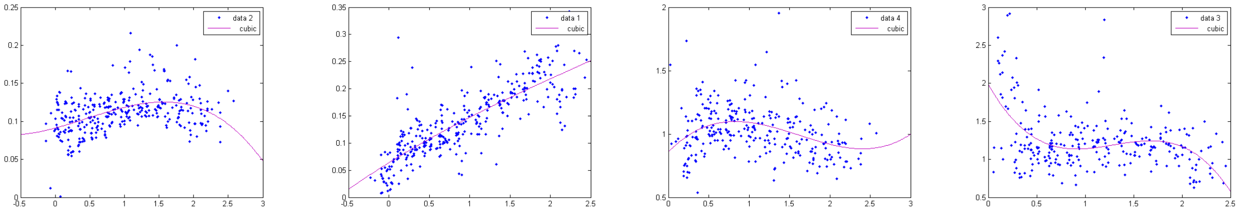


(a) Intensities (b) Filtered intensities



(c) Differences (d) Normalized differences

Figure 2: Processing the intensity signal. The output of the segmentation is indicated by green.



(a) Thickness (b) Spacing (c) Consecutive thickness (d) Consecutive spacing

Figure 3: Statistics collected from manual ribcage segmentations

As can been seen in figure 3 the thickness and the spacing are highly dependent on this vertical position, thus the permitted range for these quantities are gradually changing during the search. This graduality is assured by a cubic polynomial which is denoted by a purple curve in figure 3. Calculating the standard deviation from this polynomial gives the width of the permitted range. Values of $1\sigma < w < 2\sigma$ were proved to work well.

These constraints are incorporated into a dynamic programming problem shown in figure 4. Simplifying the problem by focusing only on one of the borders of a rib we can ignore the thickness. The vertical axis in fig. 4 represents the vertical position, the horizontal corresponds to the spacing. Thus each point on the plane defines a vertical position and a spacing value. The white region is permitted, the purple is forbidden. The possible starting and terminal points are located in the green and yellow area respectively. The aim is to find the least cost path between the green and the yellow region.

The cost for stepping on a vertical position comes from the differentiated intensities. The cost of stepping on a point with a given spacing comes from the statistics.

The cyan dots represent a path on this plane. A possible step can be calculated by moving downwards by the value of the current spacing (marked by an arc on fig. 4), and then choosing a point from a line segment crossing this point and having a $-45°$ slope. The endpoints of this line segment are determined by the consecutive spacing statistics. Each point on this line can have a different cost based on this statistics as well.

Considering both borders of the ribs we get an extra dimension, and this rectangle on the plane becomes a volume in the 3 dimensional space. In this volume the steps are alternating between the two planes. Stepping from an upper border to a lower border will take place on the $vertical position - thickness$ plane, stepping from a lower border to an upper border will be on the $verical position - spacing$ plane.

The cost of a step consists of 3 main components:

$$c = c_i + c_p + c_d \tag{1}$$

The image cost c_i is the current value at the given vertical position on the differentiated intensity signal. The density cost c_d controls the number of ribs to find. The probability cost c_p has 4 components: thickness, spacing, consecutive spacing ratio and consecutive thickness ratio.

Approximating the density functions of these quantities by a position dependent normal distribution, we can get a probability value for a given thickness, spacing or consecutive spacing at each vertical position:

$$p(x, y) = \frac{1}{\sigma}\phi\left(\frac{y - f(x)}{\sigma}\right) \tag{2}$$

Where p denotes the probability in question, x the vertical position, y the dependent quantity, f the cubic polynomial and σ the deviation.

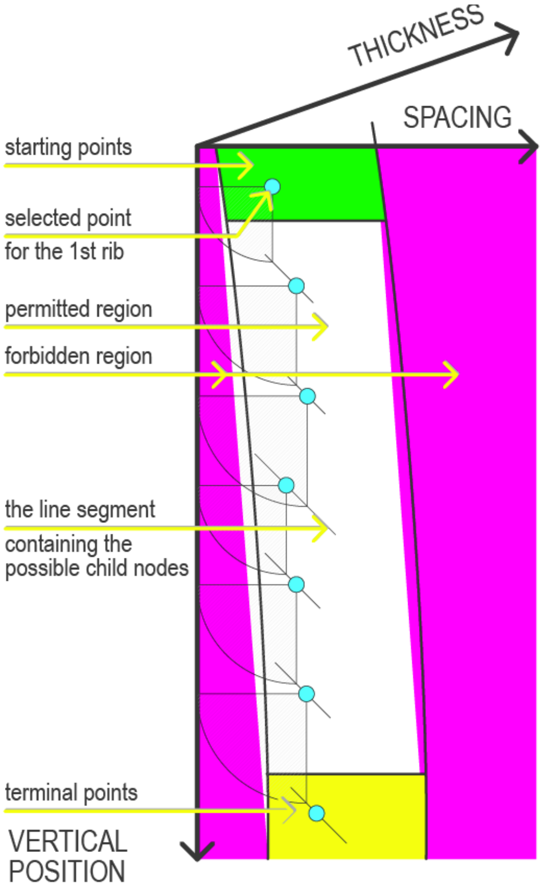


Figure 4: Search space of the dynamic programming problem

Table 1: Results. Found ribs per lungfields.

	true positive	false positive	inaccurate
previous method	7.0	0.28	0.15
proposed method without probability cost (1.2σ)	7.5	0.35	0.05
proposed method with probability cost (1.9σ)	7.2	0.06	0.03

As the dynamic programming accumulates the costs additively, the negative logarithm of the probabilities have to be used as cost. This way the mean squares of the deviations will be cumulated at the end of the path:

$$\sum_{i=1}^{N} c_p^i = -ln \prod_{i=1}^{N} p(x^i, y^i) = -ln \prod_{i=1}^{N} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(y^i - f(x^i))^2}{2\sigma^2}} = Nln(\sigma\sqrt{2\pi}) + \sum_{i=1}^{N} \frac{(y^i - f(x^i))^2}{2\sigma^2} \tag{3}$$

N is the number of ribs found, x^i is the position of the i-th rib, y^i can either be the thickness, spacing, consecutive spacing ratio of the i-th rib. The constant term can be grouped to the density cost c_d, as it depends only on the number of found ribs. The 2 in the denominator can also be left out:

$$\frac{\sum_{i=1}^{N} (y^i - f(x^i))^2}{\sigma^2} \tag{4}$$

σ^{-2} becomes the weighting factor between the different probability costs.

This probability cost has equal strength over the full signal, but the signal can be weaker at some places, that is why the aforementioned local RMS compensation was needed.

Unfortunately the cost of a path will depend on the number of nodes as well. If each cost is negative the minimum cost path search tries to break the path into more nodes to reduce the final cost. If each cost is positive the effect is the opposite. A balance have to be made by introducing a density coefficient c_d which will be added to each node's cost.

To keep down the computation time the search volume has to be coarsely quantized. The vertical resolution is set to around 600 depending on the size and shape of the lung. The resolution of the thickness and spacing axes are further reduced by a factor of two and four respectively.

III. Results

The method was evaluated on a set of 39 images independent from the ones used for creating statistics. The results are compared to a previous version where the permitted regions were set manually, and they changed linearly. This version lacked of signal normalization and probability costs as well. Introducing probability costs and statistics based permitted ranges significantly reduced false positives and raised true positives compared to the previous method, however turning off probability cost, and using tighter permitted range yields even more true positives on the expense of more false positives (see table 1). It must be noted that wider permitted range heavily slows down the procedure. The main advantage of this algorithm that its output does not requires further revision steps due to the low false positive level.

References

[1] P. de Souza, "Automatic rib detection in chest radiographs," *Computer vision, graphics, and image processing*, 23(2):129–161, 1983.

[2] S. Juhász, Á. Horváth, L. Nikházy, G. Horváth, and Á. Horváth, "Segmentation of anatomical structures on chest radiographs," *XII Mediterranean Conference on Medical and Biological Engineering and Computing 2010*, pp. 359–362, 2010.

SUPERSCALABLE MODELING SYSTEMS

Benedek IZSÓ
Advisors: István RÁTH, Dániel VARRÓ

I. Introduction

Nowadays, model-driven engineering (MDE) plays a crucial role in the development of critical embedded systems. Designer tools used for software or system development *represent data as graph*. Processing of design models involve *complex queries* expressed by graph patterns that must be evaluated on *large models*. Such operations are involved in rule-based data processing (used in expert systems), in model transformation (with model synchronization) or in the validation of design models.

In these scenarios *efficient incremental query answering* is required. This means that while a client application edits the model, match sets of (previously fixed) graph pattern queries must be available instantaneously (within milliseconds) between transactions. Transactions of typical MDE applications modify small portion of the model, while large batch-like model edits are rare. The client application must be able to efficiently load data from permanent storage, as it highly impacts initialization time.

Rete [1] is a state-saving algorithm suitable to form the basics of such a query engine. Returning matches takes less than a millisecond [2], but in exchange for speed, the cache must be created initially, it must be maintained on each modification and it must be stored. However, current implementations scale up to hundreds of thousands of elements in model size [2], in today's software development larger models come into picture, consisting ten million or more elements. To create a database that handles *large models* and offers *efficient evaluation of complex queries*, larger computers could be built (which is very costly), or as described in Sec. II. *the algorithm can be scaled up to a loosely coupled cluster*.

II. Distributed model query system

In this paper the architecture of a distributed, Rete based database is proposed, which runs on a grid of computers. It will be capable of storing large models, and evaluating complex queries efficiently.

A. Architecture

A user *application* can store or retrieve model elements, and subgraphs can be fetched matched by graph pattern queries.

The middle layer is responsible for the fast storing and retrieving of individuals, and for efficient data persistence. High performance storage technologies (like key-value stores or column-family databases) use *data shards* for the distribution of model elements. This layer should be able to send notifications to the type indexers of the distributed Rete layer about elements of their associated types.

The *distributed Rete* layer is a dataflow network, with three type of nodes. Entry nodes are the *type indexers* which sort elements (objects or relations) into sets, based on their type. These typed elements (as tokens) flow into intermediate *worker nodes* which perform operations (like filtering tokens based on constant expressions, or joining them on variables). Workers store partial results of a query in their own memory. Production nodes terminate the Rete network, consisting immediately returnable results. Connections between nodes can be *local* (within one computer) or *remote* (between computers).

The bottom layer of the database system is the *physical grid* of computers, connected by ethernet network. The database server runs on this architecture: instance data is sharded between computers, and Rete network nodes are distributed. The implementation must allow *dynamic relocation of nodes* from one computer to another and *dynamic reconfiguration of the network* to enable elastic scaling and autonomous run-time optimization.

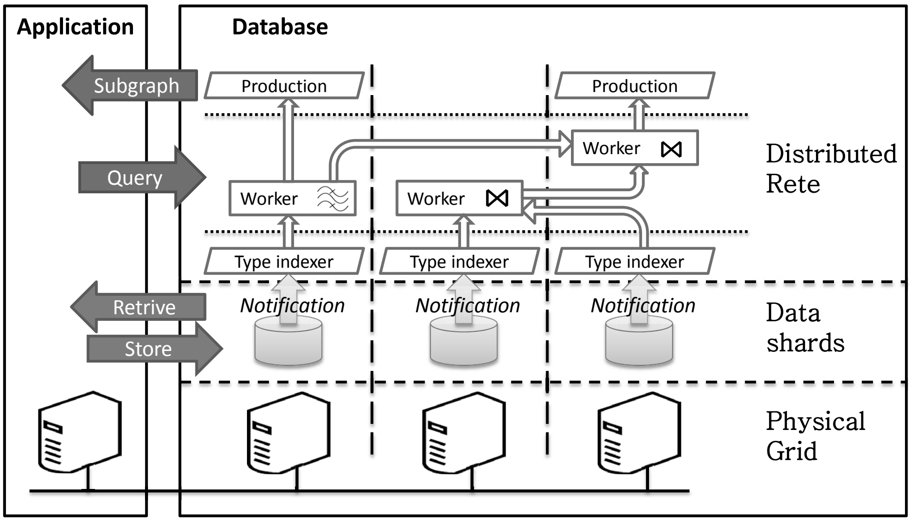


Figure 1: A distributed, Rete based model store and query system

B. *Scalability considerations*

The following guidelines should be implemented and evaluated during the prototype testing:

- Rete nodes should be in RAM, but the ones partitioned into one computer must use less memory, than physically available. When the limit is reached, *dynamic node relocation* can be used.
- Slow remote (ethernet) channels should be used, where source workers have the smallest memory. In that case token flow will be kept at minimum between computers.
- Order of Rete nodes highly affect evaluation performance. Initially heuristics for node placement based on the pattern or the initial data can be used, if it is known beforehand. At runtime, *dynamic reconfiguration of the network* allows to reconnect nodes in a different order.
- When matched subgraphs are small, it can be affordable to place the application and the database on different computers. However, placing them on one computer would enhance performance.
- The data sharding layer should sort model elements by their type and send notification to only one indexer, instead of broadcasting tokens and let type indexers to filter elements.
- Data on permanent storage should be in a compact and easily parseable format.

III. Related Work and Conclusion

Incremental, distributed query engines were proposed earlier. However they were only simulated [3], or they scaled up in the number of rules [4], and not in the number of model elements. Although, big data storages (usually based on MapReduce) provide fast object storing and individual retrieving for large models, query engines realized directly on these data structures don't provide fast, incremental query evaluation. That is, why this approach separates the sharded data store from the query engine.

As future work, we plan to address scalability and load balancing challenges of ad-hoc queries and query evolution, as well as the applications of this approach to collaborative modeling environments.

References

[1] C. Forgy, "Rete: A fast algorithm for the many pattern/many object pattern match problem," *Artificial Intelligences*, 19(1):17–37, 1982.

[2] G. Bergmann, Á. Horváth, I. Ráth, D. Varró, A. Balogh, Z. Balogh, and A. Ökrös, "Incremental evaluation of model queries over EMF models," in *Model Driven Engineering Languages and Systems*, vol. 6394 of *LNCS*. Springer, 2010.

[3] A. Acharya, M. Tambe, and A. Gupta, "Implementation of production systems on message-passing computers," *IEEE Trans. Parallel Distrib. Syst.*, 3(4):477–487, July 1992.

[4] B. Cao, J. Yin, Q. Zhang, and Y. Ye, "A mapreduce-based architecture for rule matching in production system," in *Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science*, CLOUDCOM '10, pp. 790–795, Washington, DC, USA, 2010. IEEE Computer Society.

AMBIGUOUS KNOWLEDGE CLASSIFICATION IN CONCEPTUAL GRAPHS

Tamás DEMIÁN
Advisor: András PATARICZA

I. Introduction

Unambiguous classification of knowledge is important in recognizing and eliminating any redundance (merging) or in checking consistency in knowledge bases (KB). In this paper we will review the different knowledge types of conceptual graphs (facts, rules and constraints) and discuss the boundaries in the light of this requirement. The conceptual graph[1] (CG) model is a knowledge representation formalism introduced by Sowa [2] in 1976. It is the synthesis of many works in AI (mainly existential graphs of Pierce, databases and logic, natural language processing, semantic networks). The term CG denotes the family of formalisms rooted in Sowa's seminal work.
Semantic integration is the process of merging information from (e.g. structurally, contextually) diverse sources. Classically, it distinguishes schema matching and data matching only, where knowledge classification is based on the universality. A more fine classification would facilitate the whole process.

II. Basic Conceptual Graphs

Basic conceptual graph (BG) is the fundamental member of the CG family. Its semantics is always relative to a vocabulary. The different knowledge types (facts, rules, constraints) constitute the so called BG family.

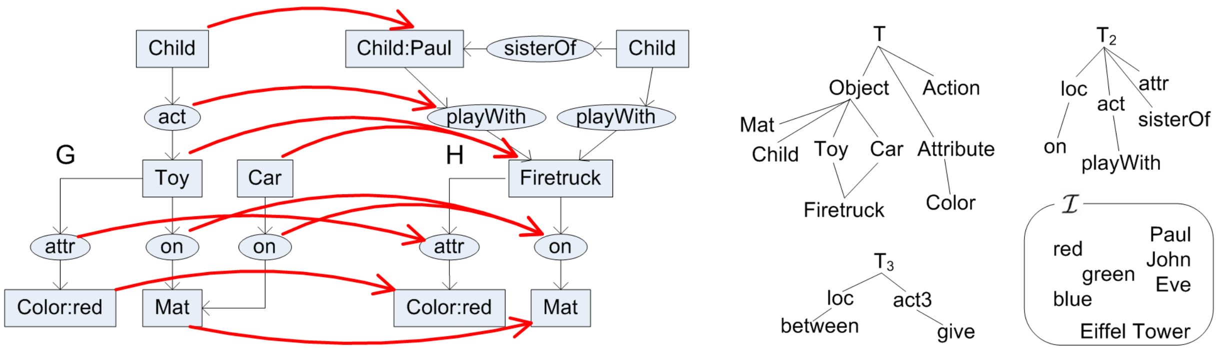


Figure 1: Two BGs with a homomorphism between them with respect to a vocabulary on the right. BGs have natural semantics. Rectangles represent concepts or individuals, ellipses represent relations. If a reletion is binary, we can order the arguments using directed edges between concept and relation nodes.

The vocabulary ($\mathcal{V}$) includes general ontological knowledge about the world (Fig. 1). Partial order in $\mathcal{V}$ is interpreted as a specialization relation. The BGs include specific factual knowledge ($\mathcal{F}$) about the world (sometimes closed existentially).

A rule expresses knowledge such as "If H is present then C can be added" where H and C are two graphs with correspondence between some of their concept nodes. H is called the hypothesis of the rule, and C its conclusion rules represent implicit general knowledge. Rules correspond to semantic entailment ($\models$) or implication ($\rightarrow$) in general. Evolution rules ($\mathcal{E}$) express transitions between worlds, inference rules ($\mathcal{R}$) express logical inference in a single world.
Constraints ($\mathcal{C}$) express a knowledge of form "if A holds",so must B", or "if A holds, B must not".

Constraints introduce an other problem, i.e., consistency ("Is the KB consistent with respect to the constraints"). Deduction from inconsistent bases is forbidden, of course. A more formal description is presented below.

Vocabulary A (BG) vocabulary is a triple $(T_C, T_R, \mathcal{I})$, where T_C is the partially ordered set of concept types and T_R is the partially ordered set of relation symbols partitioned into subsets $T_R^1, \ldots, T_R^k$ of relation symbols of arity $1, \ldots, k$, respectively. $\mathcal{I}$ is the set of individual markers. $*$ is the generic marker, and $\mathcal{M} = \mathcal{I} \cup *$ is the set of markers. T_C, T_R^i and $\mathcal{M}$ are pairwise disjoint and have a greatest element: $\top, \top_i, *$, respectively. Elements in $\mathcal{I}$ are pairwise incomparable. Any two relations with different arities are not comparable. Concept types can be seen as unary relations. The first-order logic (FOL) expression associated to $r_1(x_1, \ldots, x_n) \geq r_2(x_1, \ldots, x_n)$ is $\forall x_1, \ldots, x_n(r_2(x_1, \ldots, x_n) \to r_1(x_1, \ldots, x_n))$. The FOL semantics of the vocabulary $\Phi(\mathcal{V})$ is the conjuncion of all the corresponding FOL formulae.

BG A BG defined over a vocabulary $\mathcal{V} = (T_C, T_R, \mathcal{I})$, is a 4-tuple $G = (C, R, E, l)$ where $(C, R, E)$is a finite, undirected bipartite multigraph called the underlying graph of G. C is the concept node set, R is the relation node set, E is the family of edges.
l is a labeling function of the nodes and edges of the underlying graph that satisfies

1. $c \in C$ is labeled by a pair $(type(c), marker(c))$, where $type(c) \in T_C$ and $marker(c) \in \mathcal{I} \cup *$,
2. $r \in R$ is labeled by $l(r) \in T_R$. $l(r)$ is also called the type of r and is denoted by $type(r)$,
3. The degree of a relation node r is equal to the arity of $type(r)$,
4. Edges incident to a relation node r are totally ordered and they are labeled from 1 to $arity(type(r))$.

The FOL semantics of BGs is more complex. If a concept node does not contain individual marker then a new variable has to be assigned to it. For every relation node we assign an atomic sentence using the neighboring concept nodes as arguments. If $\phi(G)$ denotes the conjunction of formulas associated with all the concept and relation nodes then the FOL semantics of G, $\Phi(G)$ is the existential closure of of $\phi(G)$.

BG homomorphism Let G and H be two BGs defined over the same vocabulary. A homomorphism π from G to H is a mapping from C_G to C_H and from R_G to R_H, which preserves edges and may decrease concept and relation labels, that is:

- $\forall (r, i, c) \in G, (\pi(r), i, \pi(c)) \in H$,
- $\forall e \in C_G \cup R_G, l_G(\pi(e) \leq l_G(e))$.

There is a homomorphism from G to H if and only if $\Phi(\mathcal{V}), \Phi(H) \models \Phi(G)$

λ-BG A λ-BG, L=$(c_1 \ldots c_n)$G consists of a BG G and n distinguished generic concept nodes of G, $c_1 \ldots c_n$. The FOL semantics of a λ-BGis obtained from the FOL semantics of G by removing the existential qualification of all variables associated with $c_1 \ldots c_n$ (they become free variables).

λ-rule A λ-rule R is an ordered pair of λ-BGs (H, C) with a bijection between distinguished nodes (connection nodes). A rule is also denoted $R = (c_1 \ldots c_n)H \Rightarrow C$ or simply $R = H \Rightarrow C$. The FOL semantics of a rule is the universal closure of the distinguished nodes in the corresponding implication: $\Phi(R) = \forall x_1 \ldots x_n(\Phi(H) \to \Phi(C))$ where x_i is the variable corresponding to the connection node c_i.

Hom-equivalence, irredundance Two BGs G and H are said to be hom-equivalent ($G \equiv H$) if there are homomorphisms between them in both direction. Equivalence classes implied by hom-equivalence have a distinct and representative which is the "simplest" BG of the class. This element is unique up to isomorphism. A BG is called redundant if it is hom-equivalent to one of its strict subBGs. Otherwise it is called irredundant.

Constraint, satisfaction A constraint is a bicolored BG. $C_{(0)}$, the subgraph induced by the nodes of color 0, is called the condition, $C_{(1)}$, the subgraph induced by the nodes of color 1 and the frontier nodes, is called the obligation (interdiction in the case of negative constraints). A BG G satisfies a positive (resp. negative) constraint C if every (resp. no) homomorphism from the condition of C to the irredundant form of G (resp. to G) can be extended to a homomorphism from C as a whole.

Both rules and constraints can be represented with bicolored graphs or with an ordered pair of λ-BGs, but the last choice is always more expressive. Only λ-rules allow the specialization of the distinguished nodes and only λ-constraints allow signature definitions for relations. For the semantic difference between rules and constraints see Fig.2.

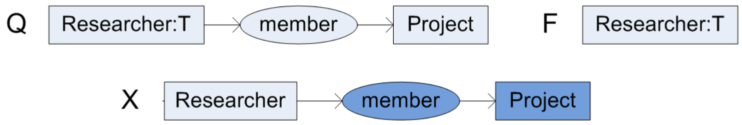


Figure 2: Example for a rule/constraint represented as a bicolored graph X. If Q is asked where X is a (rule/positive constraint/negative constraint), the answer is (yes/KB is inconsistent/no).

III. Overlapping types of the BG family

In the previous section we collected the knowledge types of the BG family: facts($\mathcal{F}$), rules ($\mathcal{R}$) and constraints ($\mathcal{C}$). General ontological knowledge shall be put into the vocabulary ($\mathcal{V}$). Now we will discuss the ambiguities amongst these knowledge types using FOL semantics.

knowledge	FOL example	prenex normal form
$\mathcal{V}$	$\forall x(Car(x) \rightarrow Object(x))$	$\forall x(Car(x) \rightarrow Object(x))$
	$\forall xy(sonOf(x,y) \rightarrow childOf(x,y))$	$\forall xy(sonOf(x,y) \rightarrow childOf(x,y))$
$\mathcal{F}$	$\exists x(Car(x) \wedge owns(Batman,x))$	$\exists x(Car(x) \wedge owns(Batman,x))$
	$Human(Batman)$	$Human(Batman)$
$\mathcal{R}$	$\forall xz(\exists y(r(x,y) \wedge r(y,z)) \rightarrow r(x,z))$	$\forall xyz(r(x,y) \wedge r(y,z) \rightarrow r(x,z))$
$\mathcal{C}^+$	$\forall x(\exists y(fatherOf(x,y)) \rightarrow Man(x))$	$\forall xy(fatherOf(x,y) \rightarrow Man(x))$
$\mathcal{C}^-$	$\forall x(Woman(x) \rightarrow \neg\exists y(fatherOf(x,y)))$	$\forall xy\neg(Woman(x) \wedge fatherOf(x,y))$

Table 1: Examples for different knowledge types. There are two different examples for types $\mathcal{V}$ and $\mathcal{F}$. Formulae of the second column come directly from the FOL semantics ($\Phi()$) of the appropriate BG constructs.

The semantic of constraints is also unique (Fig. 2), but [1] shows that if we do minor modifications on the vocabulary, for any negative constraint we can construct an equivalent positive one. Moreover, the positive and negative constraints in Tab.1 are equivalent without any vocabulary modification, if the KB knows that the extensions of $Man()$ and $Woman()$ predicates are disjunct and they completely cover the extension of $Human()$.

If a formula does not contain variables just constants, it represents obviously a factual knowledge.

However, any fact can be put in a rule with an empty hypothesis. Any knowledge in the vocabulary can also be put into a rule, where the narrower concept type will be the hypothesis, and the broader will be the conclusion. So rules are such an expressive constructs that they can express any fact or general knowledge.

The prenex form of the examples for $\mathcal{F}$ and $\mathcal{V}$ in Tab. 1 suggests that there is no transition between $\mathcal{F}$ and $\mathcal{V}$. [1] introduces the technique of flattening the vocabulary resulting an equivalent KB with an unordered vocabulary and a verbose BG. Flattening can be seen an a $\mathcal{V} \to \mathcal{R}$ transformation followed by exhaustive application of only these rules on the BG and finally by leaving of these rules. After flattening the new KB is equivalent with the old one in static sense. But if we throw both the structure (partial order) of the vocabulary and the corresponding rules then extend the KB with new nodes, this equivalence may fail. The lack of stronger equivalence is more obvious, if we consider the reverse direction. Recovering the original vocabulary from the flat one and from the BG would require logical inductions.

Finally, symbols and statements in $\mathcal{V}$ can be seen as meta-constraints on the KB. For example, if the character sequence $Spiderman \notin \mathcal{I}$ but it occurs in a BG, or an other concept node is explicitly stated as $Action$ and $Object$ in a BG (incomparable types in a coreference class), we can also say that the BG is inconsistent.

So, any kind of knowledge in the BG family can be expressed using only rules and positive constraints. It is quite easy to decide whether there is an equivalent factual ($\mathcal{F}$) or ontological knowledge for a given rule. The types of BG family are quite practical and intuitive. However, there are many knowledge types that are not supported directly or at all by the BG family. For example, a type definition $\forall x(\exists y(hasChild(x, y)) \leftrightarrow Parent(x))$ which can be decomposed to rules or constraints (Table 2) according to the structure of the corresponding BG. A relation signature can be expressed by a constraint. Since relation symbols with different arities are incomparable in the vocabulary we can not put these knowledge types into $\mathcal{V}$. Finally there are plenty of simple statements which requires negation directly. For example, using the BG family we can not express that every person is a man or a woman. Extension of BGs with negation leads to Full Conceptual Graphs (FCG), where any kind of knowledge (except for constraints) can be seen as a fact or a rule.

knowledge	BG family	prenex normal form
type definition	$\mathcal{R}$ or $\mathcal{C}$	$\forall xy(hasChild(x, y) \to Parent(x))$
	$\mathcal{R}$ or $\mathcal{C}$	$\forall x\exists y(Parent(x) \to hasChild(x, y))$
relation signature	$\mathcal{C}$	$\forall xy(act(x, y) \to Agent(x))$
subtype completeness	-	$\forall x(Human(x) \to (Man(x) \vee Woman(x)))$

Table 2: Knowledge types not covered by the BG family.

IV. Conclusion

Normal forms are important in redundance elimination or in any kind of semantic integration. There is no serious obstacle to define such a form for the BG family. Sometimes the classification is not intuitive (e.g., a type definition $\notin \mathcal{V}$) and highly relies on FOL. In other words, the BG layer on FOL is not opaque. If explicit negation is required in the KB, classification becomes indefinite beside the fact that good computational properties of BGs do not survive. So, in general case the reasonableness of this kind of classification is doubtful compared to classic first-order techniques.

References

[1] M. Chein and M.-L. Mugnier, Eds., *Graph-based Knowledge Representation - Computational Foundations of Conceptual Graphs*, Springer, 2009.

[2] J. F. Sowa, Ed., *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, 1984.

CALCULATION OF MEMBRANE CURRENTS OF NEURONS BY USING KERNEL METHODS

Dorottya CSERPÁN
Advisors: Zoltán SOMOGYVÁRI, Gábor HORVÁTH

I. Introduction

Eukariotyc cells maintain a membrane potential, which plays a vital role e.g. in signalling and transportation mechanisms. It is created by the concentration difference of various ions between the two sides of the membrane. The neurons besides the cardiac muscle cells are able to change their membrane potential significantly within a few milliseconds. This mechanism is called action potential and is generated by the change in the membrane permeability what indicates an ion current through the membrane. We are interested in those initializer of the action potential, which are caused by the neighbouring neurons, the most typical way of this is the synaptic current. A synaptic current at one point of the cell might not makes the cell to create the action potential, but the sum of more is able to do that. The goal of this paper is to present a method for determining the propagation and initialization features of the action potential based on recordings outside the neurons done by multielectrodes.

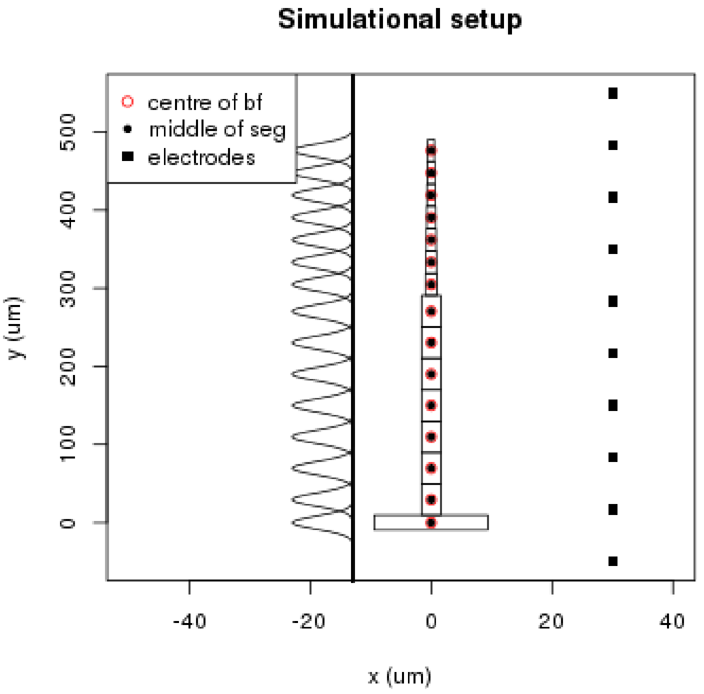


Figure 1: The simulational setup. The ballstick neuron model built up form the segments (black rectangles) is placed to the $x = 0$ position, the electrodes marked with black squares are found in a 30 μm distance. On the left side the one dimensional projection of the Gaussian base functions are shown.(bf: base function, seg: segments)

II. Overview of Relevant Current Source Density Methods (CSD)

The relationship between the current sources and the generated potentials is given by the Poisson-equation:

$$\sigma \nabla^2 \Phi(\mathbf{r}, t) = -C(\mathbf{r}, t), \tag{1}$$

where σ is the electrical conductivity of the extracellular medium, Φ is the extracellular potential, $\mathbf{r}$ refers to the position, and $C(\mathbf{r}, t) = \sum_{n=1}^{N} I_n(t)\delta^3(\mathbf{r} - \mathbf{r_n})$ is the current source density, the summation goes over all point sources,the position of the n^{th} current source is $\mathbf{r_n}$. There are several methods which use different assumptions for solving the above mentioned equation.

A. *Traditional CSD*

The traditional CSD method [1] uses the recorded extracellular potential from a laminar electrode, which is placed perpendicularly (z direction) to the layers of the cortex. Based on this setup and assuming that the layers are endless and homogeneous, the current source densities of each layer can be given:

$$C(z_j) = -\sigma \frac{\Phi(z_j + h) - 2\Phi(z_j) + \Phi(z_j - h)}{h^2} \tag{2}$$

where z_j is the position along the z-axis of the j^{th} electrode and h is the inter-electrode distance.

B. *Kernel CSD*

The kCSD method [2] extends the applicability of the traditional CSD by enabling to calculate the current sources at arbitrary positions by using kernel methods.

The CSD distribution can be expressed as the sum of the M sources:

$$C(\mathbf{x}) = \sum_{j=i}^{M} a_j \tilde{b}_j(\mathbf{x}) \tag{3}$$

$\tilde{b}$ is the source function correlated to the segments of the neuron,a_j is a multiplication constant.

The generated potential by the $\tilde{b}_i$ is denoted as b_i, and the connection is made by the A linear operator.

$$b_i(x, y, z) = A\tilde{b}_i(x, y, z) = \frac{1}{4\pi\sigma} \iiint \frac{\tilde{b}_i(x', y', z')}{\sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2}} dx'dy'dz' \tag{4}$$

There is an infinite number of source distributions we could use, one of the most common one is the Gaussian source function:

$$\tilde{b}_i(x, y, z) = e^{-\frac{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}{R}} \tag{5}$$

Here R is the double of the variance of the Gaussian function. The connection between the current sources densities and potentials is introduced by the A operator($A : \tilde{F} \rightarrow F$)

$$\Phi(\mathbf{x}) = AC(\mathbf{x}) = \sum_{i}^{M} a_i b_i(\mathbf{x}) \tag{6}$$

where $b_i = A\tilde{(b)}_i$.

The detailed derivation of this method can be found in the [2]. reference, due to the length limitation it's not detailed here. To determine the CSD distribution in arbitrary positions(x), the following kernel functions were introduced:

$$K(\mathbf{x}_k, \mathbf{x}_l) = \sum_{i=1}^{M} b_i(\mathbf{x}_k) b_i(\mathbf{x}_l) \tag{7}$$

$$\tilde{K}(\mathbf{x}_k, \mathbf{y}_l) = \sum_{j=1}^{M} b_(\mathbf{x}_k)\tilde{b}_j(\mathbf{y}_l) \tag{8}$$

Using the simulated or measured extracellular potentials (V) and assuming $\tilde{K}$ is invertible the solution for C is straightforward.

$$C(\mathbf{x}) = \tilde{\mathbf{K}}^T(\mathbf{x})\tilde{\mathbf{K}}^{-1}\mathbf{V} \tag{9}$$

C. *Spike CSD*

The Spike CSD [3] aims to calculate the current source distribution of single neurons. This requires the estimation of the cell-electrode distance and a simplified model of the shape of the neuron. Separating the potential patterns generated by different neurons is a crucial point. A key for doing this is the separation based on the clustering of potential shapes generated by the action potentials, which are like fingerprints: different for every neuron. The limitation of this model is the simplified morphology of the model and the low spatial resolution.

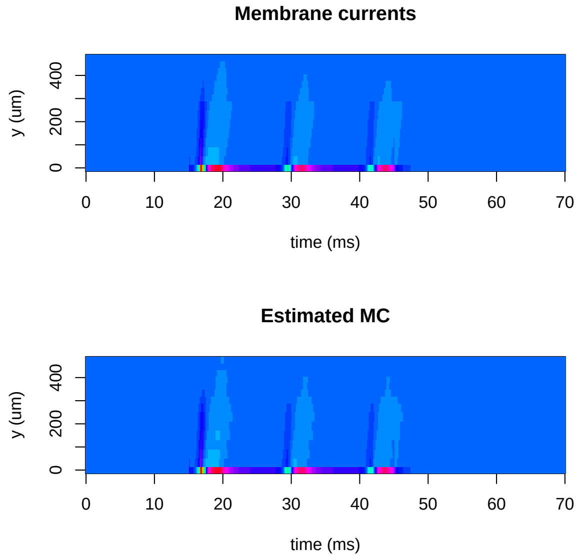


Figure 2: The original and the calculated current source distribution on colour map.

III. The kCSD method for single neurons (ksCSD)

In this method we mix the main ideas of the previous two methods, that is the calculation of current source distribution of single neurons by using kernel methods. The main advantages of this method is the more accurate spatial resolution, but for this the morphology of the cell is needed. Fluorescent dyes enable such reconstruction after the recording was done. In order to test the method simulations with different parameters were carried out.

A. *Simulation*

The simulation of extracellular potential of a ballstick neuron model was done by using LFPy [4], which is a tool designed to calculate the extracellular potential of model neurons. An excitatory current lasting 100 ms injected into the soma-ball assured the continuous action potential generation. The neuron model consisted of 15 segments with varying diameters. In the method three dimensional Gaussian source functions were used which were aligned on the axis of the model neuron.

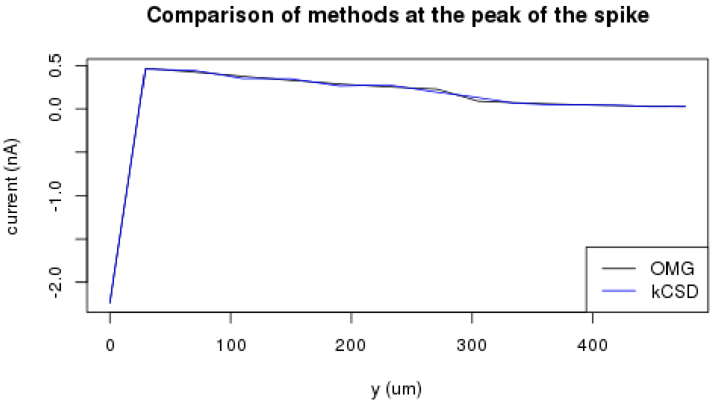


Figure 3: The shape of the current distributions (OMC: Original Membrane Current, kCSD: estimated currents) along the cell at the peak of the spike.

B. *Parameters*

In the calculation shown here the following parameters were used (Fig. 1.):

- Number of electrodes: 10
- Number of source functions: 15
- Width of source functions: 70 um
- Cell to electrode distance: 30 um

C. *Results*

The colourmaps on Figure 2. show the spatio-temporal distribution of the membrane currents used in the simulation and also the estimated ones by using the ksCSD method. The colourmaps are almost identical, but there are some small differences as Figure 3 represents. Furthermore it's important to notice, that this is just one setup of the several parameters without any noise, and neurons have a much more complex morphology. Further on we wish to use the ksCSD method on experimental data in order to understand the propagation and initialization features of the action potential.

IV. Conclusion

In this paper a novel method for calculating the current source density distribution was shown. Although additional experiments are required for establishing the morphology of the cells, with this extra information high spatial resolution is reachable. The model used in this paper is a simple ballstick model, the extension towards more complex morphology is required. The preliminary results suggest that it's worthful to continue developing this method towards more detailed neuron models.

References

[1] C. Nicholson and J. A. Freeman, "Theory of current source density analysis and determination of conductivity tensor for anuran crebellum," *J. Neurophysiol.*, 38(2):356–368, 1975.

[2] J. Potworowski, W. Jakuczun, S. Lski, and D. Wójcik, "Kernel current source density method," *Neural Comput.*, 24(2):541–575, Feb. 2012.

[3] Z. Somogyvári, D. Cserpán, I. Ulbert, and P. Erdi, "Localization of single-cell current sources based on extracellular potential patterns: the spike csd method.," *Eur J Neurosci.*, 36(10):3299–3313, Nov. 2012.

[4] H. Linden, E. Hagen, S. Leski, E. Norheim, H. Pettersen, and G. Einevoll, "Lfpy: A tool for simulation of extracellular potentials.," *Front Neuroinforn.*, (147), 2011.

MODELING OF SOLENOID ACTUATORS

Ivor Dülk
Advisor: Tamás Kovácsházy

I. Introduction

Solenoid actuators are electromechanical converters most commonly used for flow control purposes in hydraulic and pneumatic systems. In its fields of application, the so called sensorless principle is becoming more and more popular. This concept relies on an adequate model of the system, from which certain physical parameters i.e. plunger velocity and position might be estimated by measuring purely input/output electrical signals. Major advantage of the before mentioned principle is the possibility to save costly external sensors and their additional hardware "overhead", resulting in improved robustness and cost effectiveness.

My research aims to study solenoid actuators and the sensorless principle, focusing on three main areas. First, the complex structure of such devices has to be appropriately modeled, on which further estimation algorithms can be based. Secondly, measurement and control issues have to be accounted for in a sense how the system is excited and data extracted from the model. Besides, necessary actuating roles must be fulfilled as well. Since these actuators are most commonly used in embedded systems, special criteria hold for the implementation of possible models and algorithms, considering CPU effort, memory usage etc.

In technical literature dealing with the sensorless principle regarding solenoid actuators, the thermal behavior and properties of the system are not elaborately modeled or simply omitted. However, under real operating conditions the ambient and internal temperature of a solenoid can significantly change, influencing the properties of the other subsystems as well (resistance, inductance). The first year of my PhD studies has been dedicated to the thermal modeling of multilayer structures, which correspond to a model of the above mentioned actuators from the point of heat conduction. Nonetheless, a complete experimental measuring and testing environment has been designed and built intent for "real life" solenoid analyses. This setup is vital to highlight main features of these devices and to test or validate future models and algorithms. These two topics are discussed in the following chapters.

II. Discussion

In this chapter, an overview is given over the main operating principles of solenoid actuators and related research results.

A Operation of a Solenoid Actuator

Literature dealing with solenoid modeling is vast [1]-[5]. A block diagram and structural picture of a solenoid, used for flow control purposes, is given in Fig. 1. Without loss of generality, four subsystems can be identified; the electrical side converting voltage to current, the electromechanical generating magnetic force from coil current, the mechanical subsystem transferring forces to plunger position and the hydraulic one representing a hydraulic resistance through an adjustable outflow orifice. In Fig. 1 the first two subsystems are merged into one called as the electromagnetic side. From an operational point of view, current builds up in the winding due to the terminal voltage (usually from PWM). Through the air gap the magnetic force, (related to coil current) and external load (e.g. fluid pressure) immerge the plunger into the housing, thus altering the outflow orifice. These forces are counteracted by the valve return spring.

A solenoid actuator's major input parameters are voltage, external load and temperature. Strictly speaking its output quantity is hydraulic resistance (outflow orifice); however, in case of relays or force actuators (free plunger, no orifice) the hydraulic side does not even exist. Besides, it is the position of the plunger which, through only geometric constraints, directly determines the intake and outlet areas. Based on the previous explanation it seems reasonable to consider position being the characteristic output quantity of solenoid actuators. From a modeling point, a voltage (current) to plunger position transfer function forms a closed, completely described system to which a final hydraulic conversion can be simply added later. In fact, the valve only experiences a certain immerging force on the plunger formulated from fluid pressure, irrespective of other hydraulic phenomena.

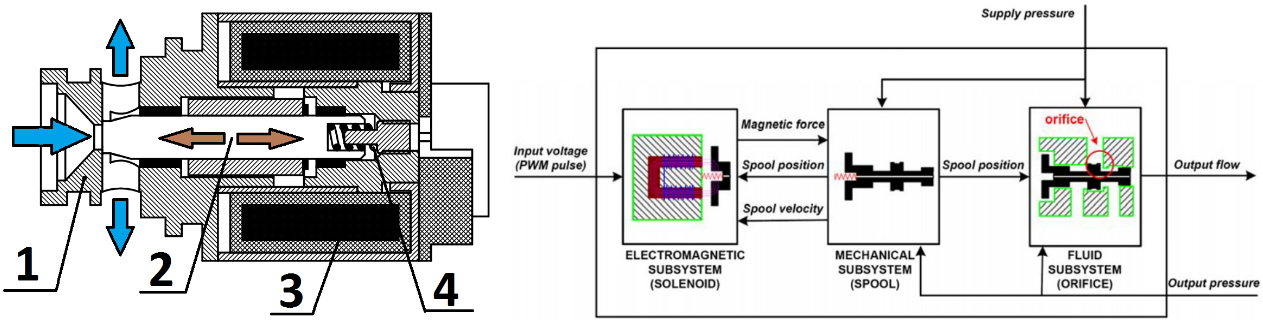


Fig. 1: Block diagram of a solenoid valve. 1: orifice, 2: plunger, 3: winding, 4: return spring. [1]

B An Experimental Setup for Solenoid Analyses

Existing solutions i.e. [3]-[5] employ LVDT (linear variable differential transformer), eddy current, laser transducer etc. and load cells for measuring position and force respectively. Position has to be measured in a range of a few millimeters to a few tens of millimeter. Additionally, the above sensors are quite expensive and require dedicated drive hardware, further increasing costs. During our analyses, a test solenoid with less than 1 mm of plunger travel has been studied, therefore special considerations had to be taken into account in position detection. We also sought for an alternate solution so as to avoid the use of fairly expensive sensors. The mechanical test rig must have met the following requirements:

- Accurate position detection in the range of 0.85 mm with high resolution,
- A way to measure and provide adjustable external force disturbances to the solenoid,
- Introduced secondary mechanical phenomena e.g. friction and hysteresis have to be minimal,
- Simple design and cost effectiveness. The test bench is intent for identification and "proof of concept" purposes instead of large scale production.

We have proposed the use of cheap reflective optical transducers, CNY70. These devices incorporate a photodiode and transistor operated at the infrared region. Necessary drive circuit is very simple and a good scalability in the measurable position range is achieved. From the point of external forces, the complete layout is arranged in a vertical configuration and the solenoid is subject to the gravitational forces of given masses. The gravitational forces being known, force sensors are not necessary in steady state, thus there is no need for load cells or other transducers.

In the final layout, special considerations had been made to reduce any subsidiary friction and hysteresis that would bias measurement results. Possible sources causing measurement uncertainty, such as machining imperfection, mechanical deformation have been also taken into account. A schematic of the experimental setup is given in Fig. 2.

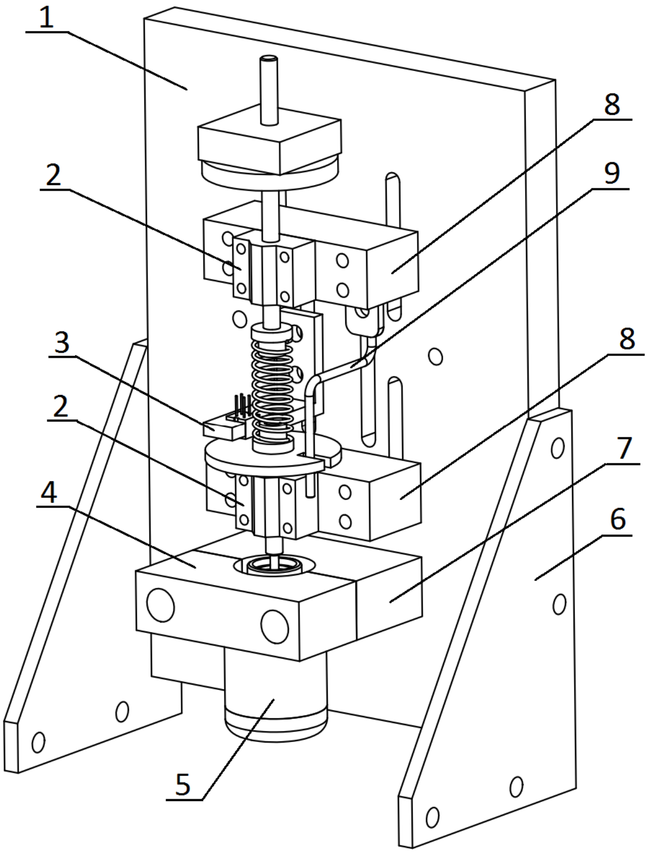


Fig. 2: Schematic of the experimental setup. 1: board, 2: linear bearing, 3: CNY70 optical sensor, 4: valve mount, 5: solenoid valve, 6: foot, 8: spacer, 9: anti-rotation rod of reflective disc.

C *Thermal Modeling of Solenoids: Concept of Multilayer Laminates*

In present sensorless methods related to solenoid actuators the thermal characteristics of the system has not been modeled [3]-[5]. In real applications, however, thermal dependence can significantly change important model parameters (resistance), thus decreasing estimation accuracy. In order to improve future models, we have been conducting research on modeling related heat phenomena. From a modeling aspect, a solenoid can be considered as a multilayered cylindrical structure, layered along the radial direction.

The underlying model is a “physical”, linear distributed system with finite, multi dimensional geometric extensions. Layers are chosen such that they form homogenous domains respectively and have their unique material and physical properties (parameter set). From the governing partial differential equations, describing heat diffusion, an analytical solution (physical model) of the system is sought, which expresses the spatial temperature distribution in every point of the studied problem. Different layers and their underlying equations are linked to each other through interface conditions e.g. continuity of heat flux. Major thermal effects to be considered are multi dimensional heat conduction, convection and internal heat generation (coil). The general model of multilayer composites is illustrated in Fig. 3, considering 2D conduction in slab shaped bodies. Slabs #1 to #N have physical lengths H and L, thermal conductivity k and internal heat generation q. At layer junctions, the model incorporates flux type boundary conditions Q, which may model thin film heating. Surface boundary conditions are convection with T ambient temperature and h convection coefficient (impedance). Subscripts L, r, u, b denote left, right, upper and bottom side respectively. Excitations are convective of transverse and lateral type (transverse is left to right, lateral is top to bottom), transverse flux at layer junctions and internal heat generation. The latter can be used to model heat generated in the coils.

In our research, an analytical solution is desired. Though FEM programs give a rapid and accurate numerical solution to this problem, the real physical processes, relations and parameter dependence are lost. Considering embedded applications evaluation of an internal FEM routine would be too consuming. Besides, analytical ones provide a more compact, closed form solution with the possibility of further "lumped" model reduction and optimization opportunities.

In technical literature, many analytical solutions and approaches exist to treat the problem of unsteady and steady heat conduction. A quite preferred one to solve the governing partial differential equations is SOV, separation of variables. This method expresses the solution in terms of eigenfunctions and eigenvalues, provided some restrictions on the boundary conditions. When using this method the major difficulty is encountered in finding eigenvalues of the system, especially when lateral excitations are also present.

In our research, we have dealt only with steady state. A solution procedure in case of only transverse sources has been established. Using superposition, we have developed a novel approach to treat lateral sources, in which these cases are traced back to the previous one consisting only transverse excitations. Thus, associated eigenproblems are greatly simplified and an algorithmic solution procedure is obtained, using only the transverse solution routine.

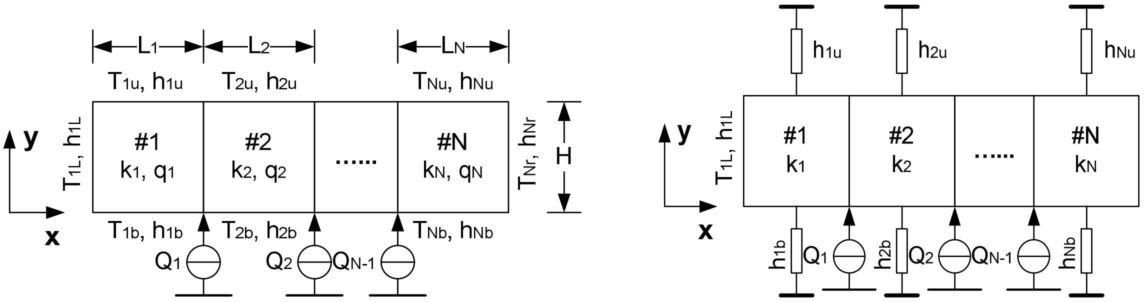


Fig. 3: Block diagram of general 2D multilayer thermal model (left) and the transverse case (right).

III. Conclusion

When using the sensorless principle measured input signals, usually electrical ones, and a model of the system serve as the basis of estimation. In present methods thermal dependence and behavior of the system is not taken into account. As an improvement possibility, we are currently working on such a model. Heat conduction in multilayer structures, corresponding to a solenoid model, are studied and a physical analytical solution is sought. Considering steady state, we have proposed a novel approach to simplify the heat conduction problem. In later analyses, a setup for measuring and testing solenoids is of paramount importance. We have also given a brief description of a novel experimental setup. Its major advantages are the novel sensing-preloading schemes it employs.

References

[1] M. Taghizadeh, A. Ghaffari, F. Najafi, "Modeling and identification of a solenoid valve for PWM control applications," Comptes Rendus Mécanique, Volume 337, Issue 3, March 2009, Pages 131-140.

[2] Richard E. Clark, Geraint W. Jewell, Stephen J. Forrest, Jan Rens, and Christophe Maerky, "Design Features for Enhancing the Performance of Electromagnetic Valve Actuation Systems," IEEE Transactions on Magnetics, vol. 41, no. 3, pp.1163-1168 March 2005.

[3] Peter Bejoum Eyabi, "Modeling and Sensorless Control of Solenoidal Actuators", PhD dissertation, Ohio State University, 2003.

[4] Ryan R. Chladny, Charles Robert Koch, and Alan F. Lynch, "Modeling Automotive Gas-Exchange Solenoid Valve Actuators," IEEE Transactions on Magnetics, vol. 41, no. 3, pp. 1155-1162, March 2005.

[5] Ryan R. Chladny and Charles R. Koch, "Flatness-Based Tracking of an Electromechanical Variable Valve Timing Actuator With Disturbance Observer Feedforward Compensation," IEEE Transactions on Control Systems Technology, vol. 16, no. 4, pp. 652-663, July 2008.